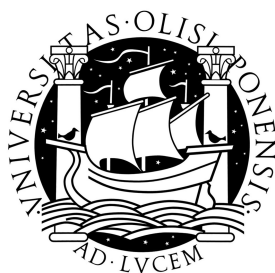


UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
Departamento de Estatística e Investigação Operacional



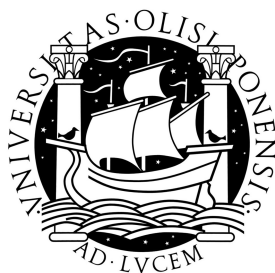
COMPARAÇÃO DE DIFERENTES ESTRATÉGIAS DE
ENGENHARIA DE TRÁFEGO EM REDES DE
TELECOMUNICAÇÕES

Rita João Santos Beato

MESTRADO EM INVESTIGAÇÃO OPERACIONAL

2011

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
Departamento de Estatística e Investigação Operacional



COMPARAÇÃO DE DIFERENTES ESTRATÉGIAS DE
ENGENHARIA DE TRÁFEGO EM REDES DE
TELECOMUNICAÇÕES

Rita João Santos Beato

Tese orientada por:

Professor Doutor Pedro Martins Pereira Serrão de Moura
(Orientador)

Professor Doutor Luís Eduardo Neves Gouveia
(Coorientador)

MESTRADO EM INVESTIGAÇÃO OPERACIONAL

2011

Resumo

A evolução dos serviços de telecomunicações tem vindo a impor novos requerimentos às redes de telecomunicações já existentes. Estas redes foram dimensionadas anteriormente baseadas num determinado padrão de procura e muitas vezes esta procura muda e o roteamento estabelecido na solução inicial pode (e deve) ser melhorado. O objectivo desta dissertação consiste em comparar e implementar diferentes estratégias de obter o melhor roteamento do tráfego numa rede, sujeito a modificações na procura.

Apresentam-se alguns modelos lineares para o problema do Desenho e Dimensionamento de uma rede e para o problema da Engenharia de Tráfego. Estes modelos são aplicados a instâncias com 25 nodos geradas aleatoriamente com diversas densidades e tráfegos.

Palavras chave: Programação Linear Inteira; Telecomunicações; Dimensionamento de redes; Engenharia de Tráfego; Roteamento.

Abstract

The evolution of telecommunications services has been imposing new requirements to existing telecommunications networks. These networks were designed based on a certain pattern of demand and this demand often changes and routing established in the initial solution can (and should) be improved. The aim of this thesis is to compare and implement different strategies to get the best routing of traffic in a network, subject to changes in demand.

Some linear models are introduced for the problem of Design and Dimensioning of a network and for the problem of Traffic Engineering. These models are applied to 25 nodes instances generated randomly with different densities and traffics.

Keywords: Integer Linear Programming; Telecommunications; Networks Dimensioning; Traffic Engineering; Routing.

Conteúdo

Resumo	iii
<i>Abstract</i>	v
1 Introdução	1
1.1 Descrição dos problemas	1
1.1.1 Desenho e Dimensionamento de uma rede	6
1.1.2 Engenharia de Tráfego	10
1.2 Estrutura da dissertação	13
2 Geração das instâncias de teste	15
2.1 Seleção dos nodos	15
2.2 Seleção das arestas	16
2.3 Seleção e disposição dos nodos de serviço (clientes)	17
2.4 Valores de tráfego gerados	17
3 Desenho e Dimensionamento de Redes de Telecomunicações	19
3.1 Modelos de Fluxos	19

3.1.1	Modelo de Fluxos Agregado	21
3.1.2	Modelo de Fluxos Desagregado	22
3.1.3	Modelo de Fluxos Escalado	23
3.1.4	Resultados Computacionais	25
4	Engenharia de Tráfego	31
4.1	Desempenho baseado na carga das ligações	31
4.1.1	Minimizar a carga média das ligações	32
4.1.2	Minimizar a carga máxima das ligações	33
4.2	Desempenho baseado na quantidade de tráfego protegido . . .	34
4.2.1	Maximizar o número de agregados com proteção	35
4.2.2	Maximizar a quantidade de tráfego com proteção . . .	36
4.3	Desempenho baseado no número de saltos de encaminhamento	36
4.3.1	Minimizar o número médio de saltos de encaminhamento	37
4.3.2	Minimizar o número máximo de saltos de encaminha- mento	38
4.3.3	Resultados Computacionais	39
5	Conclusão	43
	Bibliografia	45
	Anexos	47

Lista de Figuras

1.1	A rede original do exemplo 1.	9
1.2	Solução do desenho da rede do exemplo 1.	9
1.3	Rede já dimensionada a otimizar na Engenharia de Tráfego. .	11
3.1	Modelo de Fluxos Agregado.	21
3.2	Modelo de Fluxos Desagregado.	23
3.3	Modelo de Fluxos Desagregado Escalado.	24
4.1	Modelo para minimizar a carga média das ligações.	33
4.2	Modelo para minimizar a carga máxima das ligações.	34
4.3	Modelo para maximizar o número de agregados com proteção.	35
4.4	Modelo para minimizar o número médio de saltos de encami- nhamento.	37
4.5	Modelo para minimizar o número máximo de saltos de enca- minhamento.	38

Lista de Tabelas

1.1	Solução dos modelos Min Average e Min Max do exemplo 2. . .	12
3.1	Resultados do Dimensionamento com 5 nodos de serviço. . . .	29
3.2	Resultados do Dimensionamento com 10 nodos de serviço. . .	30
4.1	Resultados do modelo Min Average e Min Max da Engenharia de Tráfego com 5 nodos de serviço.	41
4.2	Resultados do modelo Min Average e Min Max da Engenharia de Tráfego com 10 nodos de serviço.	42
5.1	Resultados computacionais do Dimensionamento com 5 nodos de serviço.	48
5.2	Resultados computacionais do Dimensionamento com 10 no- dos de serviço.	49
5.3	Resultados computacionais do modelo Min Average da Enge- nharia de Tráfego com 5 nodos de serviço.	50
5.4	Resultados computacionais do modelo Min Average da Enge- nharia de Tráfego com 10 nodos de serviço.	51
5.5	Resultados computacionais do modelo Min Max da Engenha- ria de Tráfego com 5 nodos de serviço.	52

5.6	Resultados computacionais do modelo Min Max da Engenharia de Tráfego com 10 nodos de serviço.	53
-----	---	----

Capítulo 1

Introdução

Na primeira secção deste capítulo, faz-se o enquadramento do tema da dissertação no contexto das Redes de Telecomunicações, explicando as razões pelas quais se torna necessário estudar os modelos apresentados. É apresentado o problema do Desenho e Dimensionamento de uma rede de telecomunicações, e são introduzidos os aspectos que se prendem com a Engenharia de Tráfego. Estes aspectos são ilustrados através de dois exemplos. Por último, é descrita a estrutura desta dissertação e quais as variantes dos problemas apresentados foram estudadas.

1.1 Descrição dos problemas

As redes de telecomunicações são redes digitais usadas para enviar informação de um terminal origem para um terminal destino. Esta informação pode estar em formatos digitais (por exemplo, ficheiros) ou em formatos analógicos (por exemplo, voz ou video). Neste último caso, a informação tem de ser convertida na origem para um formato digital (através de um conversor analógico-digital) para ser enviada. No destino, recupera-se o formato origi-

nal analógico (através de um conversor digital-analógico).

Para a informação ser enviada, esta é dividida em conjuntos de *Bytes*, a que é acrescentado um cabeçalho com informação como os endereços de rede do terminal origem e do terminal destino, e outra informação de suporte, formando *pacotes de informação*. Ao conjunto de pacotes enviados pelo terminal origem ao longo do tempo para o terminal destino, chama-se *fluxo de tráfego* do terminal origem para o terminal destino. O *fluxo de tráfego* caracteriza-se pelo ritmo médio de transmissão, mais conhecido por *largura de banda*, em bits por segundo (bps), com que os pacotes são enviados ao longo do tempo. Isto representa uma rede de telecomunicações com *comutação de pacotes*.

Os componentes de uma rede de telecomunicações são os terminais/nodos de serviço (de onde parte e/ou chega informação), as ligações entre os terminais e os nodos de comutação/trânsito (pontos intermédios por onde passa a informação).

Os pacotes sofrem atrasos na rede, nomeadamente, nas ligações, onde o atraso é dado pelo tempo de transmissão do pacote mais o tempo de propagação da ligação e nos nodos de trânsito, onde o atraso se deve ao tempo de processamento do pacote (inclui leitura do cabeçalho e escolha da saída pela qual o pacote é internamente encaminhado) juntamente com o tempo na fila de espera da porta de saída. A fila de espera surge com a necessidade de acondicionar temporariamente os pacotes que chegam quase em simultâneo de diferentes portas de entrada. O tempo de espera está condicionado pela largura de banda dos pacotes e da capacidade da ligação (definida pela largura de banda máxima que a ligação consegue suportar). Como a largura de banda é a taxa de transmissão média dos pacotes de fluxo suportados pela ligação, pode haver alturas em que é maior que a capacidade da ligação, o que faz com que a fila de espera encha e, conseqüentemente, se percam alguns pacotes por falta de espaço.

No fim, tudo se resume à *qualidade de serviço* que a rede proporciona. Esta caracteriza-se por:

- o *atraso médio de cada pacote* desde que é enviado pelo terminal de origem até que seja recebido pelo terminal destino.
- o *jitter*, definido pela diferença entre o atraso máximo e o atraso mínimo de todos os pacotes de fluxo.
- a *probabilidade da perda de pacote*, dada pela percentagem de pacotes que não são recebidos no terminal destino por terem sido descartados num nodo de trânsito devido à fila de espera associada estar cheia.

O encaminhamento de uma rede de telecomunicações tem um grande peso na *qualidade de serviço* que a rede proporciona aos fluxos de tráfego suportados, e é necessário restringir o conjunto das possíveis soluções de encaminhamento. Uma das áreas a analisar para garantir os níveis de *qualidade de serviço*, é o número de saltos entre o terminal origem e o terminal destino, isto é, o número de nodos de comutação por onde os pacotes de fluxo têm de passar desde a origem até ao destino.

Como foi visto anteriormente, por cada nodo de comutação por onde o fluxo passa, há um atraso, e os tempos que mais contribuem para o atraso que os pacotes sofrem numa rede são os tempos associados a estes nodos (tempos de processamento mais tempos de fila de espera). Logo, quanto menos saltos houver entre o nodo origem e o nodo destino, menor será o atraso que o pacote sofre na rede. Neste sentido, pode ser imposto um limite máximo no número de saltos entre cada par de nodos origem/destino. Este problema é abordado em maior detalhe em [3].

Outra vantagem da redução do número de saltos, é a redução da probabilidade dos pacotes de fluxo não chegarem ao destino devido à falha de alguma componente da rede. Dado que o papel das redes de telecomunicações nas mais variadas empresas é cada vez mais indispensável, a falha de uma ligação

e consequente atraso no serviço, tem um forte impacto (negativo). Por isso mesmo, as empresas de redes de telecomunicações apostam cada vez mais na segurança do serviço, garantindo percursos de encaminhamento alternativos para reencaminhar o tráfego afectado (na totalidade ou parcialmente).

Muitas vezes as falhas acontecem em mais do que um elemento da rede ao mesmo tempo, mas é mais comum acontecer em apenas um elemento de cada vez. Os estudos realizados, quer seja pelos operadores, quer seja em variados trabalhos como esta dissertação, quase todos (se não forem mesmo todos) assumem que, caso hajam falhas, estas acontecem individualmente.

As falhas podem acontecer em qualquer um dos elementos da rede de telecomunicações. Quando acontece numa ligação, o tráfego é redireccionado por outro caminho alternativo (caso exista) que não inclua a ligação danificada. O mesmo acontece com a falha de um nodo de trânsito, no entanto os operadores podem precaver-se desta situação e instalar um segundo serviço nestes nodos para substituir em caso de falha do serviço principal. Caso a falha seja num terminal origem/destino, visto que são nodos fundamentais no percurso do tráfego, a situação é mais complicada uma vez que o fluxo tem de sair (caso seja origem) ou entrar (caso seja destino) no nodo.

Para contornar/minimizar os possíveis efeitos das falhas, as empresas de redes de telecomunicações protegem o tráfego garantindo que a rede tem percursos alternativos entre qualquer par de nodos origem e destino. Estes caminhos podem ser disjuntos apenas nas ligações (em que não há partilha de ligações mas pode haver partilha de nodos de trânsito) ou disjuntos nos nodos (não há partilha de ligações nem de nodos de trânsito).

As empresas podem ainda escolher assegurar/proteger o tráfego na sua totalidade ou apenas parcialmente através da implementação de P caminhos entre cada par de nodos origem/destino.

Proteção Total Neste caso, os vários caminhos (P) têm de ter capacidade para garantir que $(100/(P-1))\%$ do tráfego é suportada entre qualquer par de nodos de serviço. Um dos caminhos é designado de caminho de proteção e os restantes $(P-1)$ caminhos como caminhos de serviço. Estes $(P-1)$ caminhos garantem $(P-1) * (100/(P-1))\% = 100\%$ do tráfego. Isto significa que terão de ser instalados recursos para garantir o tráfego, mesmo que essas ligações não sejam usadas. No caso de falha de algum elemento num dos $P-1$ caminhos de serviço, o tráfego deste caminho $(100/(P-1))\%$ é redirecionado para o caminho de proteção. Os $P-2$ caminhos de serviço, em conjunto com o caminho de proteção, asseguram assim a totalidade do tráfego.

É comum os operadores optarem por $P = 2$, assegurando a totalidade (100%) do tráfego em cada um dos percursos alternativos (um percurso de proteção e outro de serviço). Esta opção é a que exige maior percentagem de banda larga adicional, 100% no percurso de proteção, mas custa menos aumentar a capacidade das ligações nos dois caminhos escolhidos do que adicionar ligações entre nodos onde elas não existiam para obter mais do que dois caminhos. Outra razão para a escolha de apenas dois percursos de encaminhamento com proteção total do tráfego é a desvantagem da bifurcação do tráfego por mais do que um caminho. No caso de $P = 2$ temos um caminho de serviço e um caminho de proteção para o caso de haver alguma falha no caminho de serviço, e ambos os percursos asseguram o tráfego na totalidade. Quando P é superior a 2, existem $P-1$ caminhos de serviço e 1 caminho de proteção. O tráfego é dividido pelos $P-1$ caminhos (bifurcação do tráfego) e o caminho de proteção garante a mesma percentagem de tráfego que os restantes caminhos.

Proteção Parcial Para não desperdiçar tantos recursos ou por outra razão, como por exemplo, o facto de o impacto de uma falha numa deter-

minada rede não ser tão grave, a solução pode passar por assegurar o tráfego parcialmente, resultando na divisão do tráfego por diversos caminhos (P), em que $(100/P)\%$ do tráfego é assegurado por cada caminho. Na falha de um elemento, $((P - 1) * 100/P)\%$ do tráfego está assegurado. Se $P = 2$, 50% do tráfego encontra-se assegurado; $P = 3$ resulta em 60% do tráfego protegido; no caso de $P = 4$, este valor aumenta para 75%; e assim sucessivamente. A desvantagem deste método é o aumento do *jitter*. Ao serem encaminhados por diferentes percursos, os pacotes sofrerão atrasos variados e a diferença entre o atraso máximo e o atraso mínimo (*jitter*) será muito maior do que quando os pacotes seguem pelo mesmo caminho.

O focus deste trabalho é o estudo de modelos de Desenho e Dimensionamento de redes e de Engenharia de Tráfego, onde se exige que existam dois caminhos disjuntos nos nodos entre qualquer par de nodos de serviço onde o tráfego é totalmente protegido. Estes problemas serão abordados com maior detalhe nas próximas secções.

1.1.1 Desenho e Dimensionamento de uma rede

O desenho e o dimensionamento de uma rede de telecomunicações têm como objectivo definir os elementos da rede e a rede em si:

- quais os possíveis clientes a serem servidos;
- as localizações onde poderão ser instalados os nodos para servir os clientes,;
- uma estimativa das larguras de banda que serão necessárias para o tráfego esperado entre cada par de nodos;

- as ligações que precisam de ser instaladas para servir os clientes;
- o custo de criar a rede para servir esses clientes.

A rede pode ser representada através de um grafo $G = (N, E)$, em que N representa o conjunto das possíveis localizações para os possíveis nodos de serviço e de trânsito, e E representa o conjunto das possíveis ligações a ser instaladas. O conjunto N é subdividido em dois subconjuntos disjuntos e complementares, S e T . O primeiro subconjunto é constituído pelos nodos de serviço e o segundo pelos possíveis nodos de trânsito. Um serviço entre os nodos p e q consiste em fazer chegar a q uma quantidade de tráfego t_{pq} vinda de p , com $p, q \in S$. Assume-se, sem perda de generalidade, que $p < q$ (ver [3] onde esta decisão também foi tomada).

Existem vários modelos matemáticos que podem ser utilizados para resolver este problema (e as suas possíveis variantes):

Modelo de Fluxos Agregado Como o nome indica, o fluxo segue todo agregado, sem distinção de quais as diferentes origens e destinos. Para o problema estudado nesta dissertação, com proteção total do tráfego em dois percursos de encaminhamento disjuntos nos nodos, este modelo não é aplicável. Isto acontece porque é necessário identificar qual a origem e o destino de qualquer parte do tráfego que circula na rede para assim garantir que entre qualquer par de nodos cliente existem dois percursos e que estes são disjuntos nos nodos.

Modelo de Fluxos Desagregado Neste modelo, o fluxo encontra-se desagregado, isto é, para cada quantidade de fluxo que passa numa dada ligação sabe-se qual a sua origem e qual o seu destino. Também é possível distinguir a qual dos dois caminhos entre qualquer par de nodos origem/destino pertence o fluxo. Deste modo, é possível garantir

a proteção total do fluxo por dois percursos de encaminhamento. No entanto, este modelo não garante a disjunção nos nodos dos dois percursos e, por isso, não poderá ser usado para o problema em estudo.

Modelo de Fluxos Desagregado Escalado Este modelo pode ser considerado uma particularização do modelo anterior. Aqui, em vez de se considerar o fluxo que passa numa dada aresta $\{i, j\}$ (para um dado par origem/destino $p, q \in S$), considera-se a utilização ou não da aresta $\{i, j\}$ para fazer passar todo o volume de tráfego com origem em p e destino em q , ou seja t_{pq} . Este é o modelo que será usado para Desenhar e Dimensionar as redes nesta dissertação. Este modelo já nos permite obter dois percursos de encaminhamento disjuntos nos nodos, e por isso, foi o modelo utilizado nesta dissertação.

Qualquer um destes modelos pode ser adaptado a partir dos modelos "uma origem/vários destinos" apresentados em [1], para variante "várias origens/vários destinos". No caso dos modelos com o fluxo desagregado, estes incluem capacidades nas ligações e dois percursos disjuntos nos nodos. Estes modelos são explicados com maior detalhe no Capítulo 3.

O exemplo seguinte ilustra como pode ser feito o desenho e dimensionamento de uma rede.

Exemplo 1: A Figura 1.1 apresenta a rede original, com todos os nodos de serviço (\square), os possíveis locais onde poderão ser instalados nodos de trânsito (\circ), e todas as possíveis ligações entre qualquer par de nodos, sejam de serviço (clientes) ou de trânsito. Neste caso, os nodos cliente são os nodos, 1, 4 e 7. Os serviços a serem considerados são 1-4, 1-7 e 4-7, sempre do nodo p para q com $p < q$, como foi explicado anteriormente.

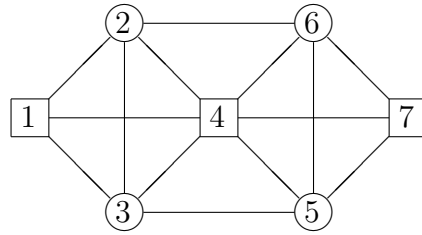


Figura 1.1: A rede original do exemplo 1.

A rede é desenhada para ter dois caminhos disjuntos nos nodos entre qualquer par de clientes e apenas são instalados os nodos de serviço e as ligações necessárias. A rede da Figura 1.2 é a rede anterior após ser dimensionada.

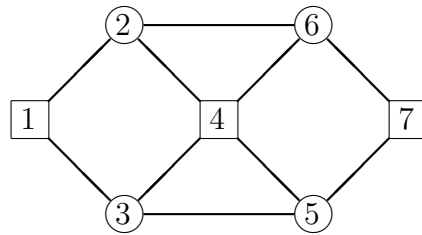


Figura 1.2: Solução do desenho da rede do exemplo 1.

Neste exemplo, todos os nodos de trânsito foram instalados mas nem todas as possíveis ligações fazem parte da solução. Os dois percursos de encaminhamento entre os nodos 1 e 4 são 1-2-4 e 1-3-4, entre os nodos 1 e 7 são 1-2-6-7 e 1-3-5-7 e finalmente, entre os nodos 4 e 7 são 4-6-7 e 4-5-7.

Admitindo, a título de exemplo de ilustração, que o tráfego entre qualquer par de clientes p, q , com $p, q \in \{1, 2, 3\}$ e $p < q$, é unitário, então a rede deverá ser dimensionada de tal forma que as ligações $\{1, 2\}$, $\{1, 3\}$, $\{6, 7\}$ e $\{5, 7\}$ possam suportar 2 unidades de fluxo e as ligações $\{2, 4\}$, $\{2, 6\}$, $\{3, 4\}$, $\{3, 5\}$, $\{4, 6\}$ e $\{4, 5\}$ possam suportar apenas 1 unidade de fluxo.

1.1.2 Engenharia de Tráfego

A Engenharia de Tráfego usa redes já desenhadas e dimensionadas, e otimiza o seu desempenho. Os aspectos que podem ser otimizados e que contribuem para o desempenho da rede são:

Cargas das ligações A carga de uma ligação é dada pelo rácio entre a soma das larguras de banda de todos os agregados que passam na ligação e a capacidade da ligação. Usualmente é um valor entre 0 e 1, mas há alturas em que a rede está *congestionada* e este quociente ultrapassa a unidade. As consequências do *congestionamento* da ligação são a perda de alguns pacotes de fluxo por não haver espaço na fila de espera ou um atraso máximo nos pacotes que não se perdem. A minimização da carga das ligações evita estas situações.

Quantidade de tráfego protegido O encaminhamento do tráfego pode ser feito de modo a maximizar a quantidade de tráfego protegido, à semelhança dos esquemas de sobrevivência falados anteriormente. A diferença é que a rede não é dimensionada para englobar o esquema de sobrevivência escolhido como acontece no Dimensionamento. Aqui a rede já se encontra dimensionada e o tráfego é que é encaminhado para assegurar uma maior quantidade de tráfego protegido, tendo em conta que para o dimensionamento pré existente, poderá não ser possível assegurar todo o tráfego.

Número de saltos no encaminhamento Os atrasos que os pacotes sofrem na rede e o *jitter* causado pelas filas de espera são proporcionais ao número de saltos que existem no percurso desde a origem ao destino, ou seja, quanto mais saltos houver, maiores são os atrasos dos pacotes e maior é o *jitter*.

Exemplo 2: Considere-se a rede da figura 1.3. Esta rede já se encontra desenhada e dimensionada com uma ligação em cada aresta. Todas as ligações têm capacidade para 1 unidade de fluxo. O tráfego entre os nodos cliente (1 – 2, 1 – 3 e 2 – 3) é de 1 unidade de fluxo, ou seja, cada ligação só pode ser utilizada por um caminho apenas entre todos os pares (p, q) .

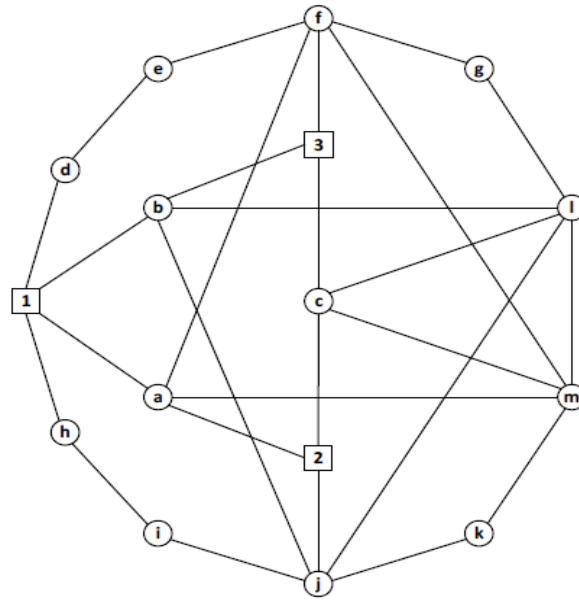


Figura 1.3: Rede já dimensionada a otimizar na Engenharia de Tráfego.

Na tabela seguinte apresentam-se as soluções dos modelos Min Average e Min Max da Engenharia de Tráfego com a rede da figura 1.3. Os dois caminhos entre cada par de nodos origem/destinho (p, q) são identificados com cores diferentes para facilitar a leitura na rede. Para cada um dos dois modelos de Engenharia de Tráfego testados são apresentados as soluções de ambos os objectivos: minimizar o número médio de saltos e minimizar o número máximo de saltos.

Solução	
Min Average	Min Max
Caminhos	
1-2:	
<ul style="list-style-type: none">• 1 - a - 2• 1 - h - i - j - 2	<ul style="list-style-type: none">• 1 - a - m - c - 2• 1 - h - i - j - 2
1-3:	
<ul style="list-style-type: none">• 1 - b - 3• 1 - d - e - f - 3	<ul style="list-style-type: none">• 1 - b - l - c - 3• 1 - d - e - f - 3
2-3:	
<ul style="list-style-type: none">• 2 - c - 3• 2 - k - j - l - g - 3	<ul style="list-style-type: none">• 2 - a - f - g - 3• 2 - k - j - b - 3
Conclusão	
Número médio de saltos = 19/6 Número máximo de saltos = 5	Número médio de saltos = 24/6 Número máximo de saltos = 4

Tabela 1.1: Solução dos modelos Min Average e Min Max do exemplo 2.

A sistematização da tabela 1.1 permite-nos analisar as duas abordagens diferentes do problema de otimização do número de saltos de encaminhamento. Quando o objectivo é minimizar a média de saltos entre os pares origem/destino, este valor é menor que na solução do modelo que tem como objectivo minimizar o número máximo de saltos num caminho. Quanto ao número máximo de saltos que qualquer caminho tem, este valor é menor no problema cujo objectivo é este mesmo: minimizar o número máximo de saltos num caminho. Que modelo usar quando se pretende otimizar o encaminhamento de uma rede relativamente ao número de saltos de encaminhamento? A resposta depende do resultado que se pretende: minimizar o custo de instalação (ter menos ligações instaladas na rede) ou maximizar a qualidade do serviço (ter o menor número de saltos em cada caminho).

1.2 Estrutura da dissertação

O processo de geração das instâncias de teste é descrito no capítulo 2. Foram consideradas no total seis redes distintas com 25 nodos, com 15%, 20% ou 25% de densidade (refletindo o número total de arestas onde poderão ser instaladas ligações) e duas quantidades de nodos de serviço para cada instância: 5 ou 10 para as redes com 25 nodos. Para cada uma destas redes, os nodos de serviço foram dispostos de quatro maneiras diferentes: na fronteira da rede, no centro, 50% na fronteira e 50% no centro e de modo aleatório. Os tráfegos considerados para cada par de nodos origem/destino foram gerados dentro de três intervalos, $[0,0.4]$, $[0,0.6]$ e $[0,1.0]$.

O seguinte passo consiste em desenhar e dimensionar as redes. No capítulo 3 são descritos três modelos que podem ser utilizados para o problema do Dimensionamento de uma rede com "várias origens/vários destinos": o Modelo de Fluxo Agregado, o Modelo de Fluxo Desagregado e o Modelo de Fluxo Desagregado Escalado. Nesta dissertação foi utilizado um Modelo de Fluxo

Desagregado Escalado, devido às necessidades de garantir dois caminhos disjuntos nos nodos entre cada par de clientes, que apenas é possível com este último modelo. No final deste capítulo apresentam-se resultados computacionais para testar a qualidade do modelo utilizado em termos da relaxação linear, face às instâncias geradas.

Finalmente, otimiza-se o encaminhamento das redes através da Engenharia de Tráfego no capítulo 4. A otimização do desempenho das redes pode basear-se nas cargas das ligações, na quantidade de tráfego protegido ou no número de saltos no encaminhamento. Foi estudado o desempenho baseado no número de saltos de encaminhamento. Os resultados computacionais apresentados no final deste capítulo baseiam-se nas redes dimensionadas/soluções obtidas com o modelo utilizado no capítulo anterior.

Por último, o capítulo 5 apresenta a conclusão de todo o trabalho realizado.

Capítulo 2

Geração das instâncias de teste

Neste capítulo descreve-se como foram geradas as redes consideradas para o Desenho e Dimensionamento das redes de Telecomunicações e, consequentemente, para os modelos de Engenharia de Tráfego, apresentados nos capítulos 3 e 4, respetivamente.

2.1 Seleção dos nodos

Usando o software MATLAB, foram criadas redes de $n = 25$ nodos que foram dispostos aleatoriamente sobre os pontos de coordenadas inteiras de grelhas bidimensionais de dimensão $[0, L] \times [0, L]$. A dimensão de cada grelha é determinada de tal forma que a dispersão dos n nodos no conjunto dos pontos de coordenadas inteiras da grelha seja igual a 1%, ou seja, $\frac{n}{(L+1)^2} = 0.01$. Assim sendo, a dimensão da grelha bidimensional usada para 25 nodos foi de $[0, 51] \times [0, 51]$.

Os n nodos são então escolhidos aleatoriamente de entre os $(L+1)^2$ pontos de coordenadas inteiras da grelha $[0, L] \times [0, L]$. É calculado um ponto fictício, designado *centro* da rede, cujas coordenadas são determinadas como sendo

a média das respectivas coordenadas de todos os n nodos dispostos aleatoriamente na rede. Determinam-se distâncias euclidianas entre qualquer par de nodos, inclusive entre todos os pontos e o *centro* da rede. As distâncias entre os nodos (excepto o *centro* da rede) serão usadas como custos no Dimensionamento da rede e na Engenharia de Tráfego.

2.2 Seleção das arestas

O passo seguinte é a escolha das arestas que fazem parte da rede. Foram consideradas três densidades: 15%, 20% e 15%. Para cada uma delas, o processo é idêntico. É necessário garantir que a rede é conexa, por isso primeiro escolhem-se arestas de modo a formar um *ciclo hamiltoniano*, ou seja, um ciclo que passa por todos os nodos uma única vez, sem repetir nodos. Porquê um *ciclo hamiltoniano* e não uma *árvore de suporte*? Para este problema não basta uma *árvore de suporte*, em que todos os nodos estão ligados entre si de modo a que haja apenas um caminho entre cada par de nodos. É necessário um *ciclo hamiltoniano* pois para o problema de Dimensionamento da Rede considerado (com proteção total do tráfego através de dois caminhos disjuntos nos nodos) são necessários dois caminhos entre cada par de nodos de serviço. Essa garantia é dada com um *ciclo hamiltoniano*, enquanto que com uma *árvore de suporte* apenas se garante um caminho.

Começa-se por escolher dois nodos da rede para serem ligados por uma aresta. Para as restantes arestas, liga-se o último nodo escolhido a um que ainda não tenha sido usado. Quando já não restarem nodos por escolher, liga-se o último nodo escolhido ao primeiro dos nodos para fechar o ciclo. As restantes arestas são escolhidas aleatoriamente até perfazer a densidade considerada, com uma única condição: o *custo* da aresta, isto é, a distância entre os extremos da aresta não pode ser superior à média das distâncias en-

tre todos os pares de nodos, excluindo o *centro* da rede. Com esta restrição, as ligações escolhidas não são demasiado longas. Esta restrição é apenas imposta após a escolha das arestas que fazem parte do *ciclo hamiltoniano* para não se restringirem demasiado as arestas que podem fazer parte do ciclo, o que poderia resultar na inexistência de uma solução para se obter uma rede conexa.

2.3 Seleção e disposição dos nodos de serviço (clientes)

Dos n nodos, apenas alguns, s , são clientes que têm de ser servidos. Para as redes com 25 nodos, temos 5 ou 10 nodos de serviço. Os s nodos de serviço foram dispostos de quatro maneiras diferentes:

- todos os nodos de serviço dispostos na fronteira da rede, isto é, sobre os nodos mais afastados do *centro* da rede;
- todos os nodos de serviço dispostos no centro da rede, isto é, sobre os nodos mais próximos do *centro* da rede;
- 50% dos nodos de serviço dispostos na fronteira da rede e 50% dos nodos de serviço dispostos no *centro* da rede;
- todos os nodos de serviço dispostos aleatoriamente sobre a rede.

2.4 Valores de tráfego gerados

Um serviço consiste em transportar um determinado tráfego de um nodo cliente p para um outro nodo cliente q , assumindo que é feito apenas num dos sentidos e que $p < q$. Os tráfegos entre os pares de nodos cliente foram

gerados aleatoriamente num intervalo $[0, \beta]$, para uma capacidade unitária em cada ligação implementada. Foram testados três intervalos de geração diferentes, com $\beta = 0.4, 0.6$ e 1.0 (ver [3], onde foi utilizada uma geração semelhante e uma capacidade também unitária em cada ligação).

Resumindo, foram geradas seis redes distintas com 25 nodos, 15%, 20% ou 25% de densidade de arestas e 5 ou 10 nodos de serviço. Para cada uma delas, os nodos de serviço foram dispostos de quatro maneiras diferentes e foram utilizados três tráfegos entre cada par $p - q$, com $p, q \in S$ e $p < q$. O *Output* deste processo é uma matriz c com os *custos* entre cada par de nodos da rede, quatro vectors v que indicam quais os nodos onde foram instalados os nodos serviço com as quatro disposições consideradas e três matrizes de tráfegos, cada uma com valores dentro dos três intervalos considerados. Estas três matrizes são simétricas por se ter assumido que os serviços são de p para q com $p < q$. Para o *custo* de cada ligação, foi considerada a distância entre os nodos terminais da ligação.

Capítulo 3

Desenho e Dimensionamento de Redes de Telecomunicações

O Desenho e Dimensionamento de uma rede de telecomunicações é explicado em maior detalhe nas secções que se seguem. Começa-se por enquadrar brevemente o problema que será estudado neste capítulo na área das Redes de Telecomunicações. Ao longo deste capítulo são introduzidos diferentes Modelos de Fluxo: o Modelo de Fluxo Agregado (secção 3.1.1), o Modelo de Fluxo Desagregado (secção 3.1.2) e o Modelo de Fluxo Escalado (secção 3.1.3). Esta última secção inclui os resultados computacionais obtidos com o Modelo de Fluxo Escalado e as instâncias geradas segundo o processo descrito no capítulo 2.

3.1 Modelos de Fluxos

O desenho de uma rede de telecomunicações é o primeiro passo quando as infra-estruturas de suporte à instalação e operação de uma rede de telecomunicações ainda não existe. Isto acontece com a criação de uma rede nova ou um serviço novo e cabe ao operador criá-la. Para isso é preciso saber

quais os possíveis clientes (nodos de serviço) que serão servidos pela rede, onde instalar os locais a que estarão ligados (nodos de trânsito), uma estimativa da largura de banda que suportará o tráfego entre todos os pares de nodos e o custo de criar a rede que servirá os potenciais clientes. A tarefa seguinte, o dimensionamento da rede, passa pela escolha das ligações que serão instaladas na rede, necessárias para servir os clientes.

O problema pode ser traduzido matematicamente por um grafo $G = (N, E)$, em que cada nodo $i \in N$ representa uma localização e cada aresta $\{i, j\}$ representa uma ligação entre os nodos i e j . Estas ligações tem uma capacidade fixa K , e nesse caso, poderá ser necessário instalar mais do que uma ligação em cada aresta. Para isso utilizam-se variáveis inteiras $u_{\{i,j\}}$ que representam o número de ligações a instalar entre i e j , cujo custo unitário é $c_{\{i,j\}}$. A instalação dos nodos de trânsito também poderá ter um custo associado. Nos modelos apresentados não se considera esta hipótese.

As localizações, N , dividem-se em dois subconjuntos: os clientes, S , e os nodos de trânsito, T , que podem ou não fazer parte da solução - $N = S \cup T$. Para qualquer par de nodos $p, q \in S$, a rede tem de suportar um fluxo de tráfego de t_{pq} unidades de tráfego. Note-se que, para efeitos desta dissertação, considerou-se que um serviço é de p para q , com $p < q$.

Para resolver este problema, existem vários modelos, entre os quais: o modelo de fluxo agregado, o modelo de fluxo desagregado e o modelo de fluxo escalado. As ligações destes modelos podem ser *unidireccionais* ou *bidireccionais*. De seguida descrevem-se, de uma forma resumida, estes modelos. Para tal, assume-se que as ligações são bidireccionais (o conjunto A inclui todos os arcos (i, j) e (j, i) se a aresta $\{i, j\} \in E$). Estes modelos baseiam-se nos modelos com uma única origem e vários destinos apresentados em [1] e foram adaptados para a variante com várias origens e vários destinos e com capacidades nas ligações.

3.1.1 Modelo de Fluxos Agregado

No Modelo de Fluxo Agregado, como o nome indica, o fluxo segue agregado, isto é, apenas se sabe o total de fluxo de tráfego que passa em cada aresta, não conhecendo as diversas origens e destinos desse tráfego. Neste modelo, para além das variáveis $u_{\{i,j\}}$ já apresentadas, utilizam-se variáveis não negativas f_{ij} que indicam a quantidade de fluxo que passa no arco (i, j) .

Este modelo apresenta-se de seguida.

$$\min \sum_{\{i,j\} \in E} c_{ij} \cdot u_{\{i,j\}} \quad (3.1)$$

$$\text{s.a.} : \sum_{(i,j) \in A} f_{ij} - \sum_{(j,i) \in A} f_{ji} = \begin{cases} \sum_{p \in S} t_{pj} - \sum_{q \in S} t_{jq} & j \in S \\ 0 & j \in N \setminus S \end{cases} \quad (3.2)$$

$$f_{ij} + f_{ji} \leq K \cdot u_{\{i,j\}} \quad \{i, j\} \in E \quad (3.3)$$

$$f_{ij} \geq 0 \quad (i, j) \in A \quad (3.4)$$

$$u_{\{i,j\}} \text{ inteiras} \quad \{i, j\} \in E \quad (3.5)$$

Figura 3.1: Modelo de Fluxos Agregado.

O objectivo deste modelo, (3.1), é minimizar os custos de criar a rede necessária a servir os clientes, neste caso, os custos de instalar as ligações necessárias. O fluxo tem de ser conservado em cada nodo ao longo do caminho, desde que sai dos nodos origem até aos nodos destino, por isso, para cada nodo de trânsito $j \in N \setminus S$, a quantidade de fluxo que sai é igual à quantidade que entra em j . Para cada nodo de serviço $j \in S$, a diferença entre a quantidade de fluxo que entra e a quantidade de fluxo que sai de j é a diferença entre a soma dos tráfegos que lhe são destinados e a soma dos tráfegos que tem de fornecer. Matematicamente, podemos representar esta condição usando o conjunto de restrições (3.2). As restrições (3.3) garantem que em cada aresta $\{i, j\}$ são instaladas ligações suficientes para assegurar

que a aresta $\{i, j\}$ tem capacidade para todo o fluxo que por ela passe. O domínio das variáveis está definido nas restantes restrições, (3.4) e (3.5).

Como neste modelo o fluxo segue todo agregado, e não se sabe qual a origem e o destino de qualquer unidade de fluxo, é impossível garantir que hajam dois percursos entre cada par de nodos origem/destino $\{p, q\}$ e que estes sejam disjuntos nos nodos. Logo, este modelo não pode ser usado para o problema em estudo.

3.1.2 Modelo de Fluxos Desagregado

Contrariamente ao modelo anterior, o Modelo de Fluxo Desagregado permite saber qual o destino das X unidades de fluxo que passam na aresta $\{i, j\}$; x_1 vão para do nodo p_1 para o nodo q_1 , x_2 vão para do nodo p_2 para o nodo q_2 , etc ..., onde $x_1 + x_2 + \dots = X$. Com isso é possível modelar dois caminhos entre cada par $p, q \in S$. Para identificar o fluxo desagregado, é necessário acrescentar novas variáveis ao modelo:

y_{ij}^{dpq} - quantidade de fluxo que passa no arco (i, j) com origem no nodo p e destino no nodo q ($p, q \in S$) no d -ésimo caminho ($d = 1, 2$).

O Modelo de Fluxos Desagregado resulta do Modelo de Fluxos Agregado, reescrito com as variáveis de fluxo desagregado. A relação entre as variáveis do Modelo de Fluxos Agregado e as novas variáveis do Modelo de Fluxos Desagregado pode ser traduzida pelas equações

$$f_{ij} = \sum_{d=1}^2 \sum_{p,q \in S} y_{ij}^{dpq} \quad (i, j) \in A \quad (3.6)$$

que indicam que a quantidade de tráfego que passa no arco (i, j) é dividida pela quantidade de tráfego entre cada um dos pares de nodos de serviço e em cada um dos percursos de encaminhamento, que passa no mesmo arco.

$$\min \sum_{\{i,j\} \in E} c_{ij} \cdot u_{\{i,j\}} \quad (3.1)$$

$$\text{s.a.} : \sum_{(i,j) \in A} y_{ij}^{dpq} - \sum_{(j,i) \in A} y_{ji}^{dpq} = \begin{cases} \sum_{p \in S} t_{pj} - \sum_{q \in S} t_{jq} & j, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S, d = 1, 2 \end{cases} \quad (3.7)$$

$$\sum_{p,q \in S} \sum_{d=1}^2 (y_{ij}^{dpq} + y_{ji}^{dpq}) \leq K \cdot u_{\{i,j\}} \quad \{i,j\} \in E \quad (3.8)$$

$$y_{ij}^{dpq} \geq 0 \quad (i,j) \in A, p, q \in S, d = 1, 2 \quad (3.9)$$

$$u_{\{i,j\}} \text{ inteiras} \quad \{i,j\} \in E \quad (3.5)$$

Figura 3.2: Modelo de Fluxos Desagregado.

Este modelo é idêntico ao modelo anterior, diferenciando apenas nas variáveis y_{ij}^{dpq} , que substituem as variáveis anteriores f_{ij} como foi já explicado. Com este modelo, garantem-se dois caminhos entre qualquer par de nodos (p, q) mas não a disjunção nos nodos dos mesmos. As variáveis deste modelo distinguem a origem e o destino das unidades de fluxo, mas ao representarem unidades de fluxo não é possível impedir que as unidades de fluxo que têm de ir da origem p para o destino q cheguem a um nodo j vindas de mais do que um único nodo i .

3.1.3 Modelo de Fluxos Escalado

Este último modelo (*MFDE*) é equivalente ao Modelo de Fluxos Desagregado, mas em vez de considerar unidades de fluxo, considera para cada par origem/destino p, q a utilização de uma dada aresta para o envio de todo o tráfego entre estes dois nodos, t_{pq} . As variáveis g_{ij}^{dpq} substituem as variáveis y_{ij}^{dpq} do modelo anterior, através das igualdades

$$g_{ij}^{dpq} = t_{pq} \cdot y_{ij}^{dpq} \quad (i,j) \in A, p, q \in S, d = 1, 2. \quad (3.10)$$

O domínio das variáveis é reduzido de $[0, t_{pq}]$ para $[0, 1]$.

$$\min \sum_{\{i,j\} \in E} c_{ij} \cdot u_{\{i,j\}} \quad (3.1)$$

$$\text{s.a. : } \sum_{(i,j) \in A} g_{ij}^{dpq} - \sum_{(j,i) \in A} g_{ji}^{dpq} = \begin{cases} -1 & j = p, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S, d = 1, 2 \\ 1 & j = q, p, q \in S, d = 1, 2 \end{cases} \quad (3.11)$$

$$\sum_{(i,j) \in A} \sum_{d=1}^2 g_{ij}^{dpq} \leq 1 \quad j \in N, j \neq q, p, q \in S \quad (3.12)$$

$$\sum_{p,q \in S} t_{pq} \cdot \sum_{d=1}^2 (g_{ij}^{dpq} + g_{ji}^{dpq}) \leq K \cdot u_{\{i,j\}} \quad \{i,j\} \in E \quad (3.13)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i,j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$u_{\{i,j\}} \text{ inteiras} \quad \{i,j\} \in E \quad (3.5)$$

Figura 3.3: Modelo de Fluxos Desagregado Escalado.

Identicamente aos modelos anteriormente apresentados, este modelo tem como objectivo, (3.1), minimizar os custos de instalação das ligações necessárias para servir os clientes. As restrições (3.11), correspondem às restrições (3.7), que garantem a conservação do fluxo ao longo da rede. Neste caso, como as variáveis g_{ij}^{dpq} tomam o valor 1 se a aresta $\{i, j\}$ é usada no d -ésimo caminho de p para q , em vez de representar unidades de fluxo como as anteriores variáveis f_{ij}^{dpq} . As restrições (3.7) garantem que se chegam f_{ij}^{dpq} unidades de fluxo da origem p com destino a q ao nodo j vindos de um nodo i no d -ésimo caminho (p, q) , têm se sair de j as mesmas f_{ij}^{dpq} unidades de fluxo da mesma origem p com o mesmo destino q para um nodo k no mesmo d -ésimo caminho (p, q) . Neste último modelo, se uma aresta $\{i, j\}$ é usada num dos dois percursos de encaminhamento entre (p, q) , então tem de haver uma outra aresta $\{j, k\}$ que também é usada no d -ésimo caminho (p, q) .

As restrições seguintes (3.12) impedem que ambos os caminhos entre qualquer par de nodos origem/destino (p, q) , utilizem arestas com um nodo em comum. Se a aresta $\{i, j\}$ é usada no caminho 1 de p para q , então não pode haver nenhuma aresta no caminho 2 de p para q que utilize os nodos i e j .

Este é um modelo com ligações de capacidade K , então a capacidade total das ligações instaladas em cada aresta tem de suportar a quantidade de fluxo que passa por esta aresta. As restrições (3.13) garantem-nos que o número de ligações instaladas em cada aresta têm de ser suficiente para suportar o fluxo todo que atravessa cada uma das arestas.

As restantes restrições, (3.14) e (3.5), definem os domínios das variáveis do modelo.

Este modelo pode ser fortalecido com a adição das restrições fortes

$$\sum_{d=1}^2 g_{ij}^{dpq} + \sum_{d=1}^2 g_{ji}^{dpq} \leq u_{\{i,j\}} \quad \{i,j\} \in E, p, q \in S. \quad (3.15)$$

Dado que as variáveis g_{ij}^{dpq} são binárias e os dois percursos de encaminhamento são disjuntos nos nodos, a restrição reduz-se a $u_{\{i,j\}} \geq 1$ se a aresta $\{i,j\}$ é utilizada por algum dos caminhos entre os nodos p e q . Esta restrição obriga a que seja instalada pelo menos uma ligação na aresta. Mas a restrição (3.13) já assegura que se a aresta é utilizada por algum caminho entre algum par de nodos origem/destino (p, q) tem de haver pelo menos uma aresta instalada, dado que as variáveis $u_{\{i,j\}}$ são inteiras. Se a aresta não for utilizada por nenhum dos caminhos de p a q , torna-se dispensável pois resulta em $u_{\{i,j\}} \geq 0$. O domínio (restrições (3.14)) garante que estas variáveis são não negativas, daí a redundância das restrições anteriores.

3.1.4 Resultados Computacionais

Nesta secção, apresentam-se os resultados obtidos com o modelo (*MDFE*) do Dimensionamento apresentado anteriormente e as instâncias geradas (ver capítulo 2).

Recordando o capítulo anterior, foram geradas instâncias com 25 nodos

no total em que 5 ou 10 são nodos de serviço. Estes nodos de serviço foram distribuídos de quatro maneiras: na fronteira da rede, no centro da rede, distribuídos aleatoriamente ou 50% dos nodos na fronteira e 50% dos nodos no centro da rede. Em relação ao número de arestas, a densidade das instâncias era 15%, 20% ou 25% em relação à rede completa em cada um dos casos.

Nas tabelas seguintes apresentam-se os resultados computacionais das instâncias testadas. Os resultados obtidos com o modelo de Dimensionamento de Fluxos Desagregados Escalado encontram-se divididos em duas tabelas, uma com os resultados das instâncias 5 nodos de serviço e outra com os resultados das instâncias 10 nodos de serviço.

Os elementos que se podem encontrar nas tabelas são:

n o número de nodos da rede (25);

s o número de nodos de serviço (5 ou 10);

d a densidade da rede face à rede completa (15%, 20% ou 25%);

Disp a disposição dos nodos de serviço (*F*, *C*, *A* ou *FF*):

F os nodos de serviço distribuídos sob os nodos mais na fronteira da rede;

C os nodos de serviço distribuídos sob os nodos mais ao centro da rede;

A os nodos de serviço distribuídos de forma aleatória;

FF 50% dos nodos de serviço distribuídos sob os nodos mais na fronteira da rede e 50% dos nodos de serviço distribuídos sob os nodos mais ao centro da rede;

t_{pq} os intervalos de tráfego considerados ($[0,0.4]$, $[0,0.6]$ ou $[0,1.0]$);

PI o tempo de CPU do modelo de Programação Inteira (em segundos);

GAP a diferença entre os valores ótimos do modelo inteiro e da sua relaxação linear (em percentagem do valor ótimo do modelo inteiro);

RL o tempo de CPU da Relaxação Linear do modelo (em segundos);

No final desta dissertação apresentam-se os valores ótimos do modelo inteiro e da sua relaxação linear, assim como os resultados que se apresentam de seguida.

Os tempos de CPU foram limitados a 9000s. Por isso para algumas instâncias não foram obtidas as soluções ótimas. Nesses casos, o tempo de CPU é 9000s e optou-se por representar o GAP entre a melhor solução obtida neste tempo e o melhor limite inferior, dado pelo programa *Xpress*. Estas instâncias estão assinaladas com um *.

Analisando as duas tabelas de resultados, verifica-se que as instâncias com 10 nodos de serviço demoram mais tempo, visto que todas as instâncias da segunda tabela atingem os 9000s sem ser obtida a solução ótima, enquanto que na primeira tabela, os tempos para se obter a solução ótima são muito inferiores. O mesmo acontece com os tempos de CPU das relaxações lineares, nas instâncias com 10 nodos de serviço, estes valores são maiores comparativamente com as instâncias com 5 nodos de serviço.

Com base nos resultados da tabela 3.1, pode-se concluir que os tempos de CPU aumentam com o aumento da densidade da rede, havendo instâncias com 25% de densidade com tempos de CPU superiores a 9000s, isto é, que não foi obtida a solução ótima. Esta conclusão também é visível nos tempos de CPU das relaxações lineares.

Na tabela 3.2, os tempos de CPU de todas as instâncias ultrapassou os

9000s, ou seja, para estas instâncias não foram obtidas as soluções ótimas e estão representados os GAPs entre a melhor solução obtida nos 9000s e o melhor limite inferior. Analisando estes valores, conclui-se que estes valores também aumentam com o aumento da densidade, isto significa que a diferença entre a melhor solução e o melhor limite inferior encontrados neste limite de tempo aumenta com o aumento da densidade. Relativamente aos tempos de CPU da relaxação linear, estes seguem a mesma conclusão: aumentam com o aumento da densidade.

Relativamente à distribuição dos nodos, com exceção das instâncias com 15% de densidade, as instâncias com os nodos de serviço na fronteira são as que têm tempos de CPU mais elevados, seguindo-se pelas instâncias com metade dos nodos de serviço instalados na fronteira e os restantes no centro. As instâncias com os nodos de serviço no centro da rede são as que têm tempos de CPU mais baixos.

Analisando agora os GAPs entre a solução inteira e a solução da relaxação linear, estes são inversamente proporcionais ao intervalo a que os tráfegos pertencem, ou seja, diminuem com o aumento do intervalo a que pertencem os tráfegos considerados. Isto significa que os limites inferiores para as instâncias com menores valores de tráfego são piores do que para as instâncias com maiores valores de tráfego. Por outro lado, pode-se verificar que os valores dos GAPs são maiores nas instâncias com maior densidade. Relativamente ao número de nodos de serviço, as instâncias com 10 nodos de serviço apresentam GAPs menores que as instâncias com 5 nodos de serviço.

Para cada uma das seis redes geradas (uma para cada densidade considerada e para cada número de nodos de serviço), a disposição dos nodos não tem influência nos valores de GAP. Na maioria dos casos os maiores valores de GAPs verificam-se nas instâncias com os nodos de serviço na fronteira da rede. No entanto, há redes em que o GAP é maior nas instâncias em que os nodos de serviço foram dispostos no centro da rede.

n	s	d	Disp	Dimensionamento			
				t_{pq}	PI	GAP	RL
25	5	15%	F	[0,0.4]	8.4s	36.1%	0.1s
				[0,0.6]	24.0s	29.7%	0.1s
				[0,1.0]	7.8s	14.4%	0.1s
			C	[0,0.4]	23.3s	34.8%	0.1
				[0,0.6]	12.2s	28.0%	0.1s
				[0,1.0]	17.8s	18.5%	0.1s
			A	[0,0.4]	23.9s	24.6%	0.3s
				[0,0.6]	30.0s	29.3%	0.1s
				[0,1.0]	192.7s	19.3%	0.1s
		20%	FF	[0,0.4]	177.8s	29.8%	0.1s
				[0,0.6]	122.9s	31.9%	0.1s
				[0,1.0]	208.4s	12.8%	0.1s
			F	[0,0.4]	1132.9s	33.7%	0.1s
				[0,0.6]	1118.8s	25.9%	0.1s
				[0,1.0]	939.8s	15.2%	0.1s
			C	[0,0.4]	28.3s	42.4%	0.2s
				[0,0.6]	107.8s	26.3%	0.1s
				[0,1.0]	93.5s	25.2%	0.1s
		25%	A	[0,0.4]	372.1s	44.2%	0.1s
				[0,0.6]	84.2s	21.9%	0.1s
				[0,1.0]	41.0s	12.7%	0.1s
			FF	[0,0.4]	53.8s	38.0%	0.3s
				[0,0.6]	54.0s	30.9%	0.1s
				[0,1.0]	242.6s	26.2%	0.1s
			F	[0,0.4]	5.44%*	52.4%	0.1s
				[0,0.6]	3.10%*	31.9%	0.1s
				[0,1.0]	567.9s	14.4%	0.1s
			C	[0,0.4]	142.9s	39.4%	0.3s
				[0,0.6]	312.4s	29.0%	0.2s
				[0,1.0]	123.7s	18.7%	0.3s
			A	[0,0.4]	2681.6s	41.4%	0.2s
				[0,0.6]	500s	29.9%	0.2s
				[0,1.0]	7603.5s	18.4%	0.3s
			FF	[0,0.4]	474.5s	46.3%	0.5s
				[0,0.6]	569.3s	23.6%	0.2s
				[0,1.0]	330.2s	16.9%	0.2s

Tabela 3.1: Resultados do Dimensionamento com 5 nodos de serviço.

n	s	d	Disp	t_{pq}	Dimensionamento		
					PI	GAP	RL
25	10	15%	F	[0,0.4]	7.08%*	21.7%	0.3s
				[0,0.6]	7.38%*	17.0%	3.3s
				[0,1.0]	5.66%*	11.2%	2.9s
			C	[0,0.4]	12.27%*	25.9%	1.1s
				[0,0.6]	4.49%*	17.1%	1.3s
				[0,1.0]	9.52%*	17.3%	2.8s
			A	[0,0.4]	10.30%*	23.6%	2.7s
				[0,0.6]	8.82%*	19.0%	2.7s
				[0,1.0]	7.60%*	15.0%	2.7s
		20%	FF	[0,0.4]	10.56%*	20.2%	0.7s
				[0,0.6]	8.40%*	20.7%	2.5s
				[0,1.0]	4.60%*	11.1%	0.3s
			F	[0,0.4]	9.06%*	22.4%	0.5s
				[0,0.6]	11.09%*	21.9%	0.7s
				[0,1.0]	4.92%*	9.1%	0.8s
			C	[0,0.4]	7.07%*	21.7%	0.1s
				[0,0.6]	11.30%*	19.8%	1.1s
				[0,1.0]	8.52%*	15.4%	0.9s
		25%	A	[0,0.4]	10.39%*	20.3%	2.3s
				[0,0.6]	6.07%*	19.1%	0.9s
				[0,1.0]	5.16%*	11.1%	1.0s
			FF	[0,0.4]	11.87%*	24.7%	1.0s
				[0,0.6]	11.20%*	18.4%	0.9s
				[0,1.0]	7.78%*	12.6%	1.0s
			F	[0,0.4]	20.58%*	32.7%	1.0s
				[0,0.6]	15.37%*	21.0%	1.1s
				[0,1.0]	14.26%*	19.5%	1.5s
			C	[0,0.4]	10.41%*	25.1%	8.4s
				[0,0.6]	15.06%*	21.2%	3.4s
				[0,1.0]	8.78%*	14.5%	2.5s
			A	[0,0.4]	15.02%*	26.8%	3.9s
				[0,0.6]	18.65%*	27.3%	3.9s
				[0,1.0]	13.68%*	19.9%	3.5s
			FF	[0,0.4]	16.84%*	26.6%	4.3s
				[0,0.6]	18.03%*	24.9%	6.4s
				[0,1.0]	12.90%*	16.4%	6.1s

Tabela 3.2: Resultados do Dimensionamento com 10 nodos de serviço.

Capítulo 4

Engenharia de Tráfego

Neste capítulo são apresentados vários modelos matemáticos para problemas de Engenharia de Tráfego (descritos no Capítulo 1). Estes modelos são baseados nos modelos apresentados em [1], aqui adaptados para várias origens e vários destinos e com dois percursos de encaminhamento.

Os diferentes modelos são utilizados para melhorar o desempenho das redes, nomeadamente a carga das ligações (secção 4.1), a quantidade de tráfego protegido (secção 4.2) e o número de saltos de encaminhamento (secção 4.3). Nesta última secção apresentam-se ainda os resultados computacionais e uma análise aos mesmos, uma vez que estes foram os modelos estudados nesta dissertação.

4.1 Desempenho baseado na carga das ligações

A carga de uma ligação é o valor que resulta de dividir a largura de banda de todos os agregados que passam pela ligação pela sua capacidade, resultando num valor entre 0 e 1. Usualmente, este valor é apresentado na forma de percentagem (entre 0% e 100%). A ligação diz-se estar *congestionada* quando

este valor é superior a 1 (ou a 100%). O resultado desta situação é a perda de alguns pacotes por a fila de espera estar sempre cheia (ou quase cheia) e o atraso dos pacotes que não são descartados ser máximo. Podemos concluir que o encaminhamento do tráfego deve ter em conta a carga das ligações para evitar ou reduzir ao máximo estas situações.

Existem duas maneiras de abordar este problema: minimizando a carga média das ligações (ou a carga total das ligações) ou minimizando a carga máxima das ligações. Porque as redes já estão dimensionadas, é necessário introduzir novos parâmetros nos modelos conhecidos, para além das variáveis já apresentadas. Estes parâmetros $b_{\{i,j\}}$ indicam a capacidade instalada na ligação $\{i,j\}$. Estas capacidades foram obtidas com os modelos para o Dimensionamento da rede e correspondem aos termos independentes das restrições (3.12), $K \cdot u_{\{i,j\}}$ após a obtenção da solução ótima nas variáveis u do modelo anterior. Note-se novamente que se assume capacidade unitária $K = 1$ em qualquer ligação instalada na aresta $\{i,j\}$.

4.1.1 Minimizar a carga média das ligações

Considerem-se o Modelo de Fluxos Desagregado Escalado (*MFDE*) e as variáveis já apresentadas. Considere-se ainda novas variáveis $v_{\{i,j\}}$ que representam a fração da capacidade da aresta $\{i,j\}$ que é utilizada.

O objectivo deste modelo pretende minimizar a carga média das ligações, minimizando as frações de capacidade instalada que todas as arestas utilizam. As restrições (4.2) garantem-nos que o encaminhamento do tráfego é feito de maneira a que o fluxo que passa na aresta $\{i,j\}$ não ultrapassa a fração utilizada da capacidade que está instalada na aresta. O domínio das novas variáveis $v_{\{i,j\}}$ é definido na restrição (4.3). As restantes restrições são idênticas aos modelos já apresentados.

$$\min \sum_{\{i,j\} \in E} v_{\{i,j\}} \quad (4.1)$$

$$\text{s.a. : } \sum_{(i,j) \in A} g_{ij}^{dpq} - \sum_{(j,i) \in A} g_{ji}^{dpq} = \begin{cases} -1 & j = p, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S, d = 1, 2 \\ 1 & j = q, p, q \in S, d = 1, 2 \end{cases} \quad (3.11)$$

$$\sum_{(i,j) \in A} \sum_{d=1}^2 g_{ij}^{dpq} \leq 1 \quad j \in N, i \neq p, p, q \in S \quad (3.12)$$

$$\sum_{p,q \in S} t_{pq} \cdot \left(\sum_{d=1}^2 g_{ij}^{dpq} + \sum_{d=1}^2 g_{ji}^{dpq} \right) \leq b_{\{i,j\}} \cdot v_{\{i,j\}} \quad \{i,j\} \in E \quad (4.2)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i, j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$0 \leq v_{\{i,j\}} \leq 1 \quad \{i,j\} \in E \quad (4.3)$$

Figura 4.1: Modelo para minimizar a carga média das ligações.

4.1.2 Minimizar a carga máxima das ligações

Uma outra solução pode passar por minimizar a carga máxima das ligações, ou seja, o maior valor de tráfego que é preciso reencaminhar em caso de falha da rede. Este valor é otimizado através da variável V , que representa a maior fração de capacidade utilizada por uma aresta. Também neste caso são considerados os parâmetros $b_{\{i,j\}}$ que indicam a capacidade instalada na aresta $\{i, j\}$.

Este modelo é semelhante ao modelo anterior, diferindo apenas na função objectivo (4.4) e nas variáveis V que utiliza. Enquanto que o modelo anterior minimiza as frações de capacidade de todas as arestas, este modelo minimiza o pior caso, isto é, a maior fração de capacidade utilizada em todas as arestas. Quanto às restrições (4.5), estas têm a mesma função que as restrições (4.2) mas em vez de o encaminhamento ser restringido pela fração de capacidade utilizada em cada aresta, é limitado pelo maior valor de fração de capacidade utilizado em todas as arestas.

$$\min V \quad (4.4)$$

$$\text{s.a. : } \sum_{(i,j) \in A} g_{ij}^{dpq} - \sum_{(j,i) \in A} g_{ji}^{dpq} = \begin{cases} -1 & j = p, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S, d = 1, 2 \\ 1 & j = q, p, q \in S, d = 1, 2 \end{cases} \quad (3.11)$$

$$\sum_{(i,j) \in A} \sum_{d=1}^2 g_{ij}^{dpq} \leq 1 \quad j \in N, i \neq p, p, q \in S \quad (3.12)$$

$$\sum_{p,q \in S} t_{pq} \cdot \left(\sum_{d=1}^2 g_{ij}^{dpq} + \sum_{d=1}^2 g_{ij}^{dpq} \right) \leq b_{\{i,j\}} \cdot V \quad \{i,j\} \in E \quad (4.5)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i, j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$0 \leq V \leq 1 \quad \{i, j\} \in E \quad (4.6)$$

Figura 4.2: Modelo para minimizar a carga máxima das ligações.

4.2 Desempenho baseado na quantidade de tráfego protegido

Os esquemas de sobrevivência referidos no Capítulo 1 também podem ser utilizados em modelos de Engenharia de Tráfego. A diferença entre estes modelos e os modelos de dimensionamento com esquemas de sobrevivência está no facto de que nestes últimos modelos a rede é desenhada e dimensionada para proteger o tráfego conforme o esquema de sobrevivência que está a ser usado. Nos modelos de Engenharia de Tráfego em que o desempenho é baseado na quantidade de tráfego protegido, as redes já existem e muitas vezes as ligações não são suficientes para proteger o tráfego na totalidade. Neste caso, tenta-se proteger o máximo de tráfego possível.

Há duas formas de abordagem destes problemas: maximizar o número de agregados com protecção ou maximizar a quantidade total de tráfego com protecção. Na primeira abordagem, essencialmente são protegidos os nodos $p, q \in S$ com menor valor de tráfego t_{pq} , maximizando o número de caminhos protegidos. Na segunda abordagem, protegem-se os nodos $p, q \in S$ com

maior valor de tráfego t_{pq} para se proteger um maior valor de tráfego. Ambas as abordagens usam variáveis binárias h^{pq} que indicam se os nodos $p, q \in S$ tem um caminho de protecção.

4.2.1 Maximizar o número de agregados com protecção

$$\max \sum_{k \in S} h^{pq} \quad (4.7)$$

$$\text{s.a.} : \sum_{(i,j) \in A} g_{ij}^{1pq} - \sum_{(j,i) \in A} g_{ji}^{pq} = \begin{cases} -1 & j = p, p, q \in S \\ 0 & j \in N \setminus S, p, q \in S \\ 1 & j = q, p, q \in S \end{cases} \quad (4.8)$$

$$\sum_{(i,j) \in A} g_{ij}^{2pq} - \sum_{(j,i) \in A} g_{ji}^{2pq} = \begin{cases} -h^{pq} & j = p, p, q \in S \\ 0 & j \in N \setminus S, p, q \in S \\ h^{pq} & j = q, p, q \in S \end{cases} \quad (4.9)$$

$$\sum_{(i,j) \in A} \sum_{d=1}^2 g_{ij}^{dpq} \leq 1 \quad j \in N, i \neq p, p, q \in S \quad (3.12)$$

$$\sum_{p,q \in S} t_{pq} \cdot \sum_{d=1}^2 (g_{ij}^{dpq} + g_{ji}^{dpq}) \leq b_{\{i,j\}} \quad \{i,j\} \in E \quad (4.10)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i,j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$h^{pq} \in \{0, 1\} \quad pq \in S \quad (4.11)$$

Figura 4.3: Modelo para maximizar o número de agregados com protecção.

Neste modelo, a função objectivo maximiza o número de nodos com um caminho de protecção. As restrições (4.8) definem os percursos de serviço para qualquer par $p, q \in S$, enquanto que as restrições (4.9) definem os percursos necessários para proteger o tráfego. Nem todos os pares (p, q) terão um encaminhamento de protecção, apenas aqueles em que $h_{pq} = 1$ no ótimo. Semelhante aos modelos apresentados anteriormente, as restrições (3.14) garantem que os percursos são disjuntos nos nodos e as restrições (4.10) asseguram

que a quantidade de tráfego que passa em cada aresta $\{i, j\}$ não ultrapassa a capacidade instalada nesta. As restantes restrições definem o domínio das variáveis.

4.2.2 Maximizar a quantidade de tráfego com protecção

Alternativamente, o modelo onde se pretende maximizar a quantidade total de tráfego protegido apenas difere deste modelo na função objectivo.

$$\max \sum_{p,q \in S} t_{pq} \cdot h^{pq} \quad (4.12)$$

Esta pretende maximizar não os nodos com caminhos de protecção, mas a quantidade de tráfego que passa por cada caminho protegido.

4.3 Desempenho baseado no número de saltos de encaminhamento

Como foi explicado no Capítulo 1, a minimização do número de saltos diminui o atraso que os pacotes sofrem na rede e o *jitter* causado pelas filas de espera e, ainda, a probabilidade de perda de pacotes de fluxo ao longo dos percursos desde a sua origem até ao seu destino. É possível optar por minimizar o número médio de saltos ou o número máximo de saltos.

O primeiro objectivo é equivalente a minimizar o número total de saltos, e no modelo são utilizadas novas variáveis w_{dpq} (para além das já apresentadas) que nos indicam o número de saltos nos caminhos entre cada origem p e cada destino q em cada um dos percursos d . Este modelo resulta numa redução no número de ligações instaladas por toda a rede.

Uma segunda hipótese é minimizar o número máximo de saltos, ou seja minimizar o pior caso. Neste caso, são utilizadas as mesmas variáveis mas apenas como auxiliares. É introduzida uma nova variável W que representa o número máximo de saltos que existem entre cada origem e cada destino em qualquer dos percursos. Este objectivo baseia-se no pior caso, ou seja, no geral os encaminhamentos da rede podem permanecer inalterados, sendo reduzido o número de saltos no pior/piores casos.

4.3.1 Minimizar o número médio de saltos de encaminhamento

Quando o objectivo é minimizar o número médio de saltos, o modelo apresentado de seguida é pode ser utilizado.

$$\min \sum_{p,q \in S} \sum_{d=1}^2 w_{dpq} \quad (4.13)$$

$$\text{s.a.: } \sum_{(i,j) \in A} g_{ij}^{dpq} - \sum_{(j,i) \in A} g_{ji}^{dpq} = \begin{cases} -1 & j = p, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S \\ 1 & j = q, p, q \in S \end{cases} \quad (3.11)$$

$$w_{dpq} = \sum_{(i,j) \in A} g_{ij}^{dpq} \quad p, q \in S, d = 1, 2 \quad (4.14)$$

$$\sum_{p,q \in S} t_{pq} \cdot \left(\sum_{d=1}^2 g_{ij}^{dpq} + \sum_{d=1}^2 g_{ij}^{dpq} \right) \leq b_{\{i,j\}} \quad \{i,j\} \in E \quad (4.10)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i, j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$w_{dpq} \geq 0 \quad p, q \in S, d = 1, 2 \quad (4.15)$$

$$(4.16)$$

Figura 4.4: Modelo para minimizar o número médio de saltos de encaminhamento.

A função objectivo (4.13) pretende minimizar o número total de saltos, o que é equivalente a minimizar o número médio de saltos, através das variáveis w_{dpq} que nos indicam o número de saltos em cada um dos dois caminhos entre p e q . As restrições que relacionam estas novas variáveis com as variáveis já usadas são as restrições (4.14). Além das restrições (4.15) que definem o domínio destas novas variáveis, as restantes restrições encontram-se nos modelos anteriores.

4.3.2 Minimizar o número máximo de saltos de encaminhamento

Se a opção for minimizar o número máximo de saltos (W), ou seja, minimizar o pior caso, o modelo usado é idêntico ao anterior.

$$\min W \quad (4.17)$$

$$\text{s.a. : } \sum_{(i,j) \in A} g_{ij}^{dpq} - \sum_{(j,i) \in A} g_{ji}^{dpq} = \begin{cases} -1 & j = p, p, q \in S, d = 1, 2 \\ 0 & j \in N \setminus S, p, q \in S \\ 1 & j = q, p, q \in S \end{cases} \quad (3.11)$$

$$\sum_{(i,j) \in A} g_{ij}^{dpq} \leq W \quad p, q \in S, d = 1, 2 \quad (4.18)$$

$$\sum_{p,q \in S} t_{pq} \cdot \left(\sum_{d=1}^2 g_{ij}^{dpq} + \sum_{d=1}^2 g_{ij}^{dpq} \right) \leq b_{\{i,j\}} \quad \{i,j\} \in E \quad (4.10)$$

$$g_{ij}^{dpq} \in \{0, 1\} \quad (i,j) \in A, p, q \in S, d = 1, 2 \quad (3.14)$$

$$W \geq 0 \quad (4.19)$$

Figura 4.5: Modelo para minimizar o número máximo de saltos de encaminhamento.

Muito semelhante ao modelo que minimiza o número médio de saltos, este modelo apenas difere nas variáveis que usa para minimizar o número máximo de saltos. Consequentemente, com exceção da função objectivo e das restrições 4.18 e 4.19, estas últimas que são idênticas às restrições 4.14 e 4.15 do modelo anterior, todas as outras restrições foram já apresentadas.

4.3.3 Resultados Computacionais

Nesta secção apresentam-se os resultados computacionais obtidos com os dois modelos de Engenharia de Tráfego que otimizam o desempenho baseado no número de saltos de encaminhamento. Ambos os modelos usam os resultados das soluções do desenho e dimensionamento da respectiva instância.

Semelhante ao dimensionamento, os resultados encontram-se divididos em duas tabelas: resultados das instâncias com 5 nodos de serviço e resultados das instâncias com 10 nodos de serviço. Nas tabelas são apresentados os tempos de CPU do modelo inteiro e da sua relaxação linear e o *GAP* entre o valor ótimo do modelo inteiro e da sua relaxação linear. No fim desta dissertação podem ser encontrados os valores ótimos, assim como os valores aqui apresentados, do modelo inteiro e da relaxação linear.

As tabelas seguintes apresentam o mesmo formato que as tabelas apresentadas no capítulo anterior.

À primeira vista é fácil comprovar que as instâncias com 10 nodos de serviço demoram muito mais tempo, comparativamente com as instâncias com 5 nodos de serviço, uma vez que na segunda tabela há algumas instâncias cujo tempo de CPU atingiu os 9000s e a solução ótima não foi obtida e outras instâncias em que não foi obtida nenhuma solução admissível no mesmo espaço de tempo. Estas instâncias estão assinaladas com * junto ao valor do GAP dado pelo programa *Xpress* no primeiro caso e ** no lugar do GAP

entre o valor da solução do modelo inteiro (inexistence nestes casos) e o valor da solução da relaxação linear no segundo caso.

Na tabela 4.1 pode-se verificar que os tempos de CPU do modelo inteiro que minimiza o número médio de saltos (Min Average) e os GAPs entre a solução do modelo inteiro e da sua relaxação linear são idênticos para todas as instâncias com 5 nodos de serviço. No entanto, na relaxação linear os tempos de CPU aumentam (não muito porque são valores muito baixos) com o aumento da densidade da rede. O mesmo acontece para as mesmas instâncias com o modelo que minimiza número máximo de saltos (Min Max). Comparando os valores de CPU entre as duas abordagens testados, estes são semelhantes, tanto para o modelo inteiro como para a relaxação linear. Relativamente aos GAPs, estes valores são consideravelmente mais baixos para as soluções do modelo Min Average, do que para o modelo Min Max. Pode-se então concluir que os limites inferiores do modelo Min Average são muito melhores do que para o modelo Min Max.

Os resultados computacionais da tabela 4.2 são claramente influenciados pela qualidade do Desenho e Dimensionamento da respectiva rede. As instâncias com 10 nodos de serviço não têm a garantia de estar a utilizar a solução ótima, uma vez que para estas instâncias a solução do problema anterior é a melhor solução que foi obtida no limite de 9000s. Como podemos verificar na tabela com os resultados de ambos os modelos utilizados na Engenharia de Tráfego, os tempos de CPU (e em alguns casos os GAPs) são muitíssimo maiores, comparativamente com as instâncias com 5 nodos de serviço. Para algumas instâncias não se chegou a obter uma solução admissível dentro do limite de tempo imposto, como já foi referido. Semelhante às conclusões para os resultados com 5 nodos de serviço, tanto os tempos de CPU do modelo inteiro como da relaxação linear, como os GAPs, apresentam valores mais elevados no modelo Min Max, comparativamente com o modelo Min Average.

n	s	d	Disp	Engenharia de Tráfego						
				t_{pq}	Min Average			Min Max		
					PI	GAP	RL	PI	GAP	RL
25	5	15%	F	[0,0.4]	0.2s	4.3%	0.0s	0.3s	11.3%	0.1s
				[0,0.6]	0.1s	2.4%	0.0s	0.3s	25.0%	0.1s
				[0,1.0]	0.2s	0.5%	0.0s	0.3s	13.9%	0.1s
			C	[0,0.4]	0.1s	1.3%	0.1s	0.1s	11.1%	0.0s
				[0,0.6]	0.2s	0.5%	0.0s	0.2s	12.5%	0.0s
				[0,1.0]	0.2s	0.7%	0.0s	0.2s	13.5%	0.0s
			A	[0,0.4]	0.3s	8.8%	0.0s	0.3s	39.9%	0.0s
				[0,0.6]	0.3s	2.9%	0.0s	0.3s	27.8%	0.1s
				[0,1.0]	0.3s	8.2%	0.0s	0.3s	16.7%	0.0s
		20%	FF	[0,0.4]	0.2s	44.0%	0.1s	0.2s	14.3%	0.1s
				[0,0.6]	0.2s	0.0%	0.0s	0.2s	14.3%	0.1s
				[0,1.0]	0.2s	3.5%	0.0s	0.2s	14.3%	0.0s
			F	[0,0.4]	0.3s	4.6%	0.0s	0.3s	32.2%	0.1s
				[0,0.6]	0.2s	2.2%	0.0s	0.5s	10.0%	0.0s
				[0,1.0]	0.3s	3.9%	0.1s	0.2s	16.7%	0.0s
			C	[0,0.4]	0.3s	2.4%	0.0s	0.2s	16.7%	0.0s
				[0,0.6]	0.2s	4.3%	0.1s	0.1s	8.3%	0.1s
				[0,1.0]	0.2s	1.9%	0.0s	0.2s	8.3%	0.0s
		25%	A	[0,0.4]	0.2s	0.0%	0.1s	0.2s	20.0%	0.0s
				[0,0.6]	0.3s	12.5%	0.0s	0.2s	20.0%	0.1s
				[0,1.0]	0.2s	20.2%	0.1s	0.3s	38.2%	0.1s
			FF	[0,0.4]	0.2s	2.3%	0.1s	0.2s	0.0%	0.1s
				[0,0.6]	0.3s	0.0%	0.1s	0.2s	0.0%	0.1s
				[0,1.0]	0.2s	0.9%	0.1s	0.2s	16.7%	0.0s
			F	[0,0.4]	0.2s	0.0%	0.1s	0.2s	16.7%	0.0s
				[0,0.6]	0.2s	5.0%	0.0s	0.3s	8.3%	0.0s
				[0,1.0]	0.2s	9.0%	0.1s	0.3s	28.6%	0.1s
			C	[0,0.4]	0.3s	7.5%	0.1s	0.3s	25.0%	0.1s
				[0,0.6]	0.2s	5.4%	0.1s	0.2s	25.0%	0.1s
				[0,1.0]	0.3s	9.0%	0.2s	0.3s	10.0%	0.1s
			A	[0,0.4]	0.3s	5.3%	0.1s	0.3s	28.6%	0.1s
				[0,0.6]	0.2s	4.7%	0.1s	0.3s	35.7%	0.1s
				[0,1.0]	0.4s	5.8%	0.1	0.3s	25.0%	0.1s
			FF	[0,0.4]	0.3s	1.4%	0.1s	0.2s	21.4%	0.1s
				[0,0.6]	0.3s	1.2%	0.1s	0.2s	7.8%	0.1s
				[0,1.0]	0.3s	0.8%	0.1s	0.2s	8.5%	0.1s

Tabela 4.1: Resultados do modelo Min Average e Min Max da Engenharia de Tráfego com 5 nodos de serviço.

				Engenharia de Tráfego						
					Min Average			Min Max		
n	s	d	Disp	t_{pq}	PI	GAP	RL	PI	GAP	RL
25	10	15%	F	[0,0.4]	391.0s	2.0%	0.2s	761.2s	14.3%	0.1s
				[0,0.6]	51.46%*	3.4%	0.3s	9000,0s	**	0.5s
				[0,1.0]	743.0s	4.6%	0.5s	3629.8s	14.3%	0.7s
			C	[0,0.4]	3.7s	2.5%	0.3s	460.0s	25.0%	0.5s
				[0,0.6]	6.6s	3.4%	0.3s	238.9s	21.1%	0.7s
				[0,1.0]	8.8s	3.8%	0.8s	10.3s	25.0%	0.6s
			A	[0,0.4]	7.1s	4.0%	0.3s	14.29%*	14.3%	0.5s
				[0,0.6]	129.2s	5.6%	0.3s	14.29%*	21.4%	0.4s
				[0,1.0]	4.5s	4.4%	0.6s	14.29%*	21.4%	1.0s
			FF	[0,0.4]	24.8s	3.8%	0.3s	14.29%*	28.6%	0.5s
				[0,0.6]	5.8s	2.3%	1.0s	10.4s	16.7%	0.8s
				[0,1.0]	9.5s	4.4%	0.1s	677.5s	18.8%	0.1s
		20%	F	[0,0.4]	510.5s	5.1%	0.1s	1980.8s	16.7%	0.2s
				[0,0.6]	9.5s	3.6%	0.1s	22.6s	16.7%	0.1s
				[0,1.0]	2.1s	4.3%	0.1s	14.29%*	21.4%	0.4s
			C	[0,0.4]	1.3s	3.5%	0.2s	1.1s	8.3%	0.1s
				[0,0.6]	4.0s	5.5%	0.1s	10.9s	30.9%	0.2s
				[0,1.0]	2.0s	3.4%	0.1s	14.29%*	28.6%	0.6s
			A	[0,0.4]	2.1s	3.6%	0.3s	5.4s	8.3%	0.3s
				[0,0.6]	1.8s	4.4%	0.1s	28.5s	8.3%	0.1s
				[0,1.0]	1.3s	4.9%	0.1s	2.3s	8.3%	0.1s
			FF	[0,0.4]	10.2s	3.8%	0.1s	2.9s	0.0%	0.2s
				[0,0.6]	3.4s	4.7%	0.1s	3246.6s	17.2%	0.2s
				[0,1.0]	30.6s	7.7%	0.1s	9000.0s	**	0.5s
		25%	F	[0,0.4]	11.1s	2.3%	0.1s	48.9s	25.0%	0.2s
				[0,0.6]	170.3s	6.1%	0.3s	9000.0s	**	0.6s
				[0,1.0]	1545.8s	8.7%	0.2s	9000.0s	**	0.7s
			C	[0,0.4]	2.9s	0.6%	0.4s	1.3s	12.5%	0.3s
				[0,0.6]	1.8s	3.5%	0.3s	1.5s	0.0%	0.4s
				[0,1.0]	1.0s	6.9%	0.1s	1.6s	0.0%	0.1s
			A	[0,0.4]	1.7s	3.0%	0.4s	105.8s	10.0%	0.4s
				[0,0.6]	8.1s	3.1%	0.6s	129.6s	0.0%	0.5s
				[0,1.0]	2.7s	4.7%	0.6s	4.1s	20.0%	0.4s
			FF	[0,0.4]	2.2s	2.0%	0.7s	3.2s	10.0%	0.5s
				[0,0.6]	13.2s	5.3%	0.5s	4.8s	10.0%	0.5s
				[0,1.0]	200.9s	7.5%	0.4s	167.2s	20.0%	0.4s

Tabela 4.2: Resultados do modelo Min Average e Min Max da Engenharia de Tráfego com 10 nodos de serviço.

Capítulo 5

Conclusão

Neste trabalho pretendia-se comparar diferentes estratégias de Engenharia de Tráfego em redes de Telecomunicações. Para tal foram abordados dois problemas da área das redes de Telecomunicações: o Desenho e Dimensionamento de uma rede e a Engenharia de Tráfego. Para cada um destes problemas foram referidos vários modelos.

Para o estudo destes dois problemas foram geradas instâncias com 25 nodos e consideradas duas quantidades de nodos de serviço, 5 e 10. Estes nodos foram distribuídos sobre os nodos da rede de quatro maneiras diferentes: todos os nodos de serviço instalados sobre os nodos mais na fronteira da rede, sobre os nodos mais no centro da rede, metade dos nodos de serviço sobre os nodos mais na fronteira da rede e os restantes sobre os nodos mais ao centro da rede ou de forma aleatória. Quanto ao número de arestas, foram geradas redes com 15%, 20% e 25% de densidade. Foram ainda considerados três intervalos para gerar os valores de tráfego. Estes foram $[0,0.4]$, $[0,0.6]$ e $[0,1.0]$.

Para o Desenho e Dimensionamento de uma rede foram apresentados primeiro um modelo de Fluxos Agregado para uma variante mais simples de apenas um caminho entre uma origem e um destino, um modelo de Fluxos

Desagregado com dois caminhos entre cada origem e cada destino e um modelo de Fluxos Desagregado Escalado com dois caminhos disjuntos nos nodos. Este último modelo foi o modelo utilizado para Desenhar e Dimensionar as instâncias geradas. Neste modelo, as ligações tinham capacidade $K=1$ e pretendia-se proteger totalmente o tráfego. Com base nos resultados obtidos, pode concluir-se que os tempos de CPU necessários à obtenção da solução ótima é proporcional ao aumento da densidade da rede.

No problema de Engenharia de tráfego usaram-se as redes resultantes do Desenho e Dimensionamento das instâncias geradas. Também para este problema se apresentaram diferentes áreas a serem estudadas, nomeadamente, minimização da carga das ligações, maximização do tráfego protegido e minimização do número de saltos de encaminhamento. Para cada uma foram apresentados dois modelos com diferentes objectivos. A área escolhida para ser estudada nesta dissertação foi a minimização do número de saltos de encaminhamento, e dois modelos cujos objectivos eram minimizar o número médio de saltos e minimizar o número máximo de saltos. Como a Engenharia de Tráfego utiliza as soluções do primeiro problema (Desenho e Dimensionamento), os resultados são influenciados pela qualidade das redes. Como se viu anteriormente, os tempos de CPU das redes cuja solução ótima foi obtida dentro do limite de tempo imposto são muito inferiores em relação às instâncias com 10 nodos de serviço.

Como trabalho futuro fica estudar o impacto que a variação do tráfego tem no encaminhamento ótimo obtido com estes modelos.

Bibliografia

- [1] GOUVEIA, Luís, MOURA, Pedro, PATRÍCIO, Pedro e SOUSA, Amaro de, *Modelação e Resolução de Problemas de Optimização em Redes de Telecomunicações*, Texto de apoio à disciplina de Redes e Telecomunicações do Mestrado de Investigação Operacional, Faculdade de Ciências da Universidade de Lisboa, 2011.
- [2] GOUVEIA, Luís, PATRÍCIO, Pedro e SOUSA, Amaro de, *Models for Optimal Survivable Routing with a Minimum Number of Hops: Comparing Disaggregated with Aggregated Models*, Working Paper, Centro de Investigação Operacional, 2009.
- [3] PATRÍCIO, Pedro Ferrão, *Modelos de Programação Linear Inteira para Problemas de Dimensionamento e Engenharia de Tráfego de Redes de Telecomunicações*, Tese de Doutoramento em Estatística e Investigação Operacional, Especialidade: Optimização, Faculdade de Ciências da Universidade de Lisboa, 2009.
- [4] OETIKER, Tobias, PARTL, Hubert, HYNA, Irene e SCHLEGL, Elisabeth, *The not so short introduction to L^AT_EX 2_ε*, Tradução portuguesa por Alberto Simões, 2004.
- [5] ANDRADE, Lenimar Nunes de, *Breve Introdução ao L^AT_EX 2_ε*, versão 2.1, Departamento de Matemática da Universidade Federal da Paraíba, 2000.

- [6] ESTRELA, Edite, SOARES, Maria Almira, LEITÃO, Maria José, *Saber Escrever Uma Tese e Outros Textos*, 5ª Edição, Publicações Dom Quixote, 2006.

Anexos

n	s	d	Disp	t_{pq}	Dimensionamento				
					<i>PI</i>		<i>RL</i>		
					VO	CPU	Gap	VO	CPU
25	5	15%	<i>F</i>	[0,0.4]	387	8.4s	36.1%	247.3	0.1s
				[0,0.6]	489	24.0s	29.7%	343.8	0.1s
				[0,1.0]	905	7.8s	14.4%	774.8	0.1s
			<i>C</i>	[0,0.4]	329	23.3s	34.8%	214.5	0.1
				[0,0.6]	396	12.2s	28.0%	285.3	0.1s
				[0,1.0]	584	17.8s	18.5%	476.2	0.1s
			<i>A</i>	[0,0.4]	351	23.9s	24.6%	264.5	0.3s
				[0,0.6]	505	30.0s	29.3%	356.8	0.1s
				[0,1.0]	726	192.7s	19.3%	585.5	0.1s
		20%	<i>FF</i>	[0,0.4]	385	177.8s	29.8%	270.4	0.1s
				[0,0.6]	452	122.9s	31.9%	307.9	0.1s
				[0,1.0]	828	208.4s	12.8%	721.9	0.1s
			<i>F</i>	[0,0.4]	406	1132.9s	33.7%	269.1	0.1s
				[0,0.6]	470	1118.8s	25.9%	348.1	0.1s
				[0,1.0]	639	939.8s	15.2%	541.9	0.1s
			<i>C</i>	[0,0.4]	175	28.3s	42.4%	100.9	0.2s
				[0,0.6]	302	107.8s	26.3%	222.5	0.1s
				[0,1.0]	312	93.5s	25.2%	233.3	0.1s
		25%	<i>A</i>	[0,0.4]	238	372.1s	44.2%	132.8	0.1s
				[0,0.6]	394	84.2s	21.9%	307.9	0.1s
				[0,1.0]	474	41.0s	12.7%	413.6	0.1s
			<i>FF</i>	[0,0.4]	210	53.8s	38.0%	130.1	0.3s
				[0,0.6]	297	54.0s	30.9%	205.3	0.1s
				[0,1.0]	381	242.6s	26.2%	281.1	0.1s
			<i>F</i>	[0,0.4]	224	5.44%*	52.4%	106.6	0.1s
				[0,0.6]	343	3.10%*	31.9%	233.5	0.1s
				[0,1.0]	511	567.9s	14.4%	437.5	0.1s
			<i>C</i>	[0,0.4]	154	142.9s	39.4%	93.4	0.3s
				[0,0.6]	216	312.4s	29.0%	153.3	0.2s
				[0,1.0]	268	123.7s	18.7%	217.9	0.3s
			<i>A</i>	[0,0.4]	225	2681.6s	41.4%	131.9	0.2s
				[0,0.6]	293	500s	29.9%	205.2	0.2s
				[0,1.0]	438	7603.5s	18.4%	357.5	0.3s
			<i>FF</i>	[0,0.4]	215	474.5s	46.3%	115.4	0.5s
				[0,0.6]	339	569.3s	23.6%	259.1	0.2s
				[0,1.0]	511	330.2s	16.9%	424.9	0.2s

Tabela 5.1: Resultados computacionais do Dimensionamento com 5 nodos de serviço.

n	s	d	Disp	t_{pq}	Dimensionamento				
					PI			RL	
					VO	CPU	Gap	VO	CPU
25	10	15%	F	[0,0.4]	1092	7.08%*	21.7%	855.2	0.3s
				[0,0.6]	1547	7.38%*	17.0%	1283.7	3.3s
				[0,1.0]	2639	5.66%*	11.2%	2343.3	2.9s
			C	[0,0.4]	690	12.27%*	25.9%	511.0	1.1s
				[0,0.6]	873	4.49%*	17.1%	723.7	1.3s
				[0,1.0]	1259	9.52%*	17.3%	1041.1	2.8s
			A	[0,0.4]	783	10.30%*	23.6%	598.2	2.7s
				[0,0.6]	1194	8.82%*	19.0%	966.6	2.7s
				[0,1.0]	1566	7.60%*	15.0%	1330.6	2.7s
			FF	[0,0.4]	1002	10.56%*	20.2%	799.5	0.7s
				[0,0.6]	1245	8.40%*	20.7%	987.8	2.5s
				[0,1.0]	1646	4.60%*	11.1%	1463.2	0.3s
		20%	F	[0,0.4]	905	9.06%*	22.4%	702.2	0.5s
				[0,0.6]	1216	11.09%*	21.9%	949.7	0.7s
				[0,1.0]	1965	4.92%*	9.1%	1785.7	0.8s
			C	[0,0.4]	509	7.07%*	21.7%	398.6	0.1s
				[0,0.6]	753	11.30%*	19.8%	603.7	1.1s
				[0,1.0]	1058	8.52%*	15.4%	894.9	0.9s
			A	[0,0.4]	603	10.39%*	20.3%	480.6	2.3s
				[0,0.6]	713	6.07%*	19.1%	576.7	0.9s
				[0,1.0]	1179	5.16%*	11.1%	1048.6	1.0s
			FF	[0,0.4]	729	11.87%*	24.7%	548.7	1.0s
				[0,0.6]	1168	11.20%*	18.4%	953.6	0.9s
				[0,1.0]	1646	7.78%*	12.6%	1438.5	1.0s
		25%	F	[0,0.4]	664	20.58%*	32.7%	447.1	1.0s
				[0,0.6]	1156	15.37%*	21.0%	912.7	1.1s
				[0,1.0]	1510	14.26%*	19.5%	1215.3	1.5s
			C	[0,0.4]	238	10.41%*	25.1%	178.2	8.4s
				[0,0.6]	426	15.06%*	21.2%	335.8	3.4s
				[0,1.0]	500	8.78%*	14.5%	427.5	2.5s
			A	[0,0.4]	539	15.02%*	26.8%	394.6	3.9s
				[0,0.6]	797	18.65%*	27.3%	579.1	3.9s
				[0,1.0]	1135	13.68%*	19.9%	908.9	3.5s
			FF	[0,0.4]	471	16.84%*	26.6%	345.6	4.3s
				[0,0.6]	650	18.03%*	24.9%	487.9	6.4s
				[0,1.0]	1087	12.90%*	16.4%	908.5	6.1s

Tabela 5.2: Resultados computacionais do Dimensionamento com 10 nodos de serviço.

n	s	d	Disp	t_{pq}	Engenharia de Tráfego				
					Min Average				
					PI		Gap	RL	
					VO	CPU		VO	CPU
25	5	15%	F	[0,0.4]	55	0.2s	4.3%	52.6	0.0s
				[0,0.6]	46	0.1s	2.4%	44.9	0.0s
				[0,1.0]	47	0.2s	0.5%	46.8	0.0s
			C	[0,0.4]	78	0.1s	1.3%	77.0	0.1s
				[0,0.6]	73	0.2s	0.5%	72.6	0.0s
				[0,1.0]	76	0.2s	0.7%	75.5	0.0s
			A	[0,0.4]	79	0.3s	8.8%	72.0	0.0s
				[0,0.6]	78	0.3s	2.9%	75.7	0.0s
				[0,1.0]	71	0.3s	8.2%	65.2	0.0s
		20%	FF	[0,0.4]	75	0.2s	44.0%	42.0	0.1s
				[0,0.6]	64	0.2s	0.0%	64	0.0s
				[0,1.0]	71	0.2s	3.5%	68.5	0.0s
			F	[0,0.4]	77	0.3s	4.6%	73.4	0.0s
				[0,0.6]	63	0.2s	2.2%	73.4	0.0s
				[0,1.0]	74	0.3s	3.9%	71.1	0.1s
			C	[0,0.4]	60	0.3s	2.4%	58.5	0.0s
				[0,0.6]	52	0.2s	4.3%	49.7	0.1s
				[0,1.0]	52	0.2s	1.9%	51.0	0.0s
		25%	A	[0,0.4]	46	0.2s	0.0%	46.0	0.1s
				[0,0.6]	52	0.3s	12.5%	45.5	0.0s
				[0,1.0]	60	0.2s	20.2%	47.9	0.1s
			FF	[0,0.4]	50	0.2s	2.3%	48.8	0.1s
				[0,0.6]	54	0.3s	0.0%	54.0	0.1s
				[0,1.0]	57	0.2s	0.9%	56.5	0.1s
			F	[0,0.4]	66	0.2s	0.0%	66.0	0.1s
				[0,0.6]	76	0.2s	5.0%	72.2	0.0s
				[0,1.0]	73	0.2s	9.0%	66.4	0.1s
			C	[0,0.4]	51	0.3s	7.5%	47.2	0.1s
				[0,0.6]	45	0.2s	5.4%	42.6	0.1s
				[0,1.0]	54	0.3s	9.0%	49.2	0.2s
			A	[0,0.4]	76	0.3s	5.3%	72.0	0.1s
				[0,0.6]	64	0.2s	4.7%	61.0	0.1s
				[0,1.0]	67	0.4s	5.8%	63.1	0.1
			FF	[0,0.4]	66	0.3s	1.4%	65.1	0.1s
				[0,0.6]	60	0.3s	1.2%	59.3	0.1s
				[0,1.0]	55	0.3s	0.8%	54.6	0.1s

Tabela 5.3: Resultados computacionais do modelo Min Average da Engenharia de Tráfego com 5 nodos de serviço.

n	s	d	Disp	t_{pq}	Engenharia de Tráfego				
					Min Average				
					PI		Gap	RL	
					VO	CPU		VO	CPU
25	10	15%	F	[0,0.4]	334	391.0s	2.0%	327.4	0.2s
				[0,0.6]	312	51.46%*	3.4%	301.3	0.3s
				[0,1.0]	316	743.0s	4.6%	301.4	0.5s
			C	[0,0.4]	237	3.7s	2.5%	231.0	0.3s
				[0,0.6]	245	6.6s	3.4%	236.6	0.3s
				[0,1.0]	257	8.8s	3.8%	247.1	0.8s
			A	[0,0.4]	281	7.1s	4.0%	269.8	0.3s
				[0,0.6]	277	129.2s	5.6%	261.5	0.3s
				[0,1.0]	269	4.5s	4.4%	257.3	0.6s
			FF	[0,0.4]	296	24.8s	3.8%	286.6	0.3s
				[0,0.6]	283	5.8s	2.3%	276.5	1.0s
				[0,1.0]	301	9.5s	4.4%	287.8	0.1s
		20%	F	[0,0.4]	267	510.5s	5.1%	253.4	0.1s
				[0,0.6]	264	9.5s	3.6%	254.5	0.1s
				[0,1.0]	285	2.1s	4.3%	272.9	0.1s
			C	[0,0.4]	232	1.3s	3.5%	223.9	0.2s
				[0,0.6]	225	4.0s	5.5%	212.6	0.1s
				[0,1.0]	242	2.0s	3.4%	233.7	0.1s
			A	[0,0.4]	235	2.1s	3.6%	226.7	0.3s
				[0,0.6]	244	1.8s	4.4%	233.4	0.1s
				[0,1.0]	237	1.3s	4.9%	225.3	0.1s
			FF	[0,0.4]	244	10.2s	3.8%	234.7	0.1s
				[0,0.6]	249	3.4s	4.7%	237.4	0.1s
				[0,1.0]	246	30.6s	7.7%	227.0	0.1s
		25%	F	[0,0.4]	278	11.1s	2.3%	271.7	0.1s
				[0,0.6]	278	170.3s	6.1%	261.1	0.3s
				[0,1.0]	284	1545.8s	8.7%	259.3	0.2s
			C	[0,0.4]	165	2.9s	0.6%	164.0	0.4s
				[0,0.6]	195	1.8s	3.5%	188.2	0.3s
				[0,1.0]	203	1.0s	6.9%	189.1	0.1s
			A	[0,0.4]	218	1.7s	3.0%	211.6	0.4s
				[0,0.6]	222	8.1s	3.1%	215.2	0.6s
				[0,1.0]	210	2.7s	4.7%	200.2	0.6s
			FF	[0,0.4]	226	2.2s	2.0%	221.4	0.7s
				[0,0.6]	217	13.2s	5.3%	205.4	0.5s
				[0,1.0]	220	200.9s	7.5%	203.5	0.4s

Tabela 5.4: Resultados computacionais do modelo Min Average da Engenharia de Tráfego com 10 nodos de serviço.

n	s	d	Disp	t_{pq}	Engenharia de Tráfego				
					Min Max				
					PI		Gap	RL	
					VO	CPU		VO	CPU
25	5	15%	F	[0,0.4]	5	0.3s	11.3%	4.4	0.1s
				[0,0.6]	4	0.3s	25.0%	3.0	0.1s
				[0,1.0]	4	0.3s	13.9%	3.4	0.1s
			C	[0,0.4]	9	0.1s	11.1%	8.0	0.0s
				[0,0.6]	8	0.2s	12.5%	7.0	0.0s
				[0,1.0]	9	0.2s	13.5%	7.8	0.0s
			A	[0,0.4]	8	0.3s	39.9%	4.8	0.0s
				[0,0.6]	9	0.3s	27.8%	6.5	0.1s
				[0,1.0]	6	0.3s	16.7%	5.0	0.0s
		20%	FF	[0,0.4]	7	0.2s	14.3%	6.0	0.1s
				[0,0.6]	7	0.2s	14.3%	6.0	0.1s
				[0,1.0]	7	0.2s	14.3%	6.0	0.0s
			F	[0,0.4]	8	0.3s	32.2%	5.4	0.1s
				[0,0.6]	5	0.5s	10.0%	4.5	0.0s
				[0,1.0]	6	0.2s	16.7%	5.0	0.0s
			C	[0,0.4]	6	0.2s	16.7%	5.0	0.0s
				[0,0.6]	6	0.1s	8.3%	5.5	0.1s
				[0,1.0]	6	0.2s	8.3%	5.5	0.0s
		25%	A	[0,0.4]	5	0.2s	20.0%	4.0	0.0s
				[0,0.6]	5	0.2s	20.0%	4.0	0.1s
				[0,1.0]	6	0.3s	38.2%	3.7	0.1s
			FF	[0,0.4]	4	0.2s	0.0%	4.0	0.1s
				[0,0.6]	4	0.2s	0.0%	4.0	0.1s
				[0,1.0]	6	0.2s	16.7%	5.0	0.0s
			F	[0,0.4]	6	0.2s	16.7%	5.0	0.0s
				[0,0.6]	6	0.3s	8.3%	5.5	0.0s
				[0,1.0]	7	0.3s	28.6%	5.0	0.1s
			C	[0,0.4]	4	0.3s	25.0%	3.0	0.1s
				[0,0.6]	4	0.2s	25.0%	3.0	0.1s
				[0,1.0]	5	0.3s	10.0%	4.5	0.1s
			A	[0,0.4]	7	0.3s	28.6%	5.0	0.1s
				[0,0.6]	7	0.3s	35.7%	4.5	0.1s
				[0,1.0]	6	0.3s	25.0%	4.5	0.1s
			FF	[0,0.4]	7	0.2s	21.4%	5.5	0.1s
				[0,0.6]	5	0.2s	7.8%	4.6	0.1s
				[0,1.0]	5	0.2s	8.5%	4.6	0.1s

Tabela 5.5: Resultados computacionais do modelo Min Max da Engenharia de Tráfego com 5 nodos de serviço.

n	s	d	Disp	t_{pq}	Engenharia de Tráfego				
					Min Max				
					PI		Gap	RL	
					VO	CPU		VO	CPU
25	10	15%	F	[0,0.4]	7	761.2s	14.3%	6.0	0.1s
				[0,0.6]	No Sol	9000,0s	**	6.5	0.5s
				[0,1.0]	7	3629.8s	14.3%	6.0	0.7s
			C	[0,0.4]	6	460.0s	25.0%	4.5	0.5s
				[0,0.6]	7	238.9s	21.1%	5.5	0.7s
				[0,1.0]	6	10.3s	25.0%	4.5	0.6s
			A	[0,0.4]	7	14.29%*	14.3%	6.0	0.5s
				[0,0.6]	7	14.29%*	21.4%	5.5	0.4s
				[0,1.0]	7	14.29%*	21.4%	5.5	1.0s
			FF	[0,0.4]	7	14.29%*	28.6%	5.0	0.5s
				[0,0.6]	6	10.4s	16.7%	5.0	0.8s
				[0,1.0]	8	677.5s	18.8%	6.5	0.1s
		20%	F	[0,0.4]	6	1980.8s	16.7%	5.0	0.2s
				[0,0.6]	6	22.6s	16.7%	5.0	0.1s
				[0,1.0]	7	14.29%*	21.4%	5.5	0.4s
			C	[0,0.4]	6	1.1s	8.3%	5.5	0.1s
				[0,0.6]	6	10.9s	30.9%	4.1	0.2s
				[0,1.0]	7	14.29%*	28.6%	5.0	0.6s
			A	[0,0.4]	6	5.4s	8.3%	5.5	0.3s
				[0,0.6]	6	28.5s	8.3%	5.5	0.1s
				[0,1.0]	6	2.3s	8.3%	5.5	0.1s
			FF	[0,0.4]	5	2.9s	0.0%	5.0	0.2s
				[0,0.6]	5	3246.6s	17.2%	4.1	0.2s
				[0,1.0]	No Sol	9000.0s	**	4.0	0.5s
		25%	F	[0,0.4]	6	48.9s	25.0%	4.5	0.2s
				[0,0.6]	No Sol	9000.0s	**	4.5	0.6s
				[0,1.0]	No Sol	9000.0s	**	4.1	0.7s
			C	[0,0.4]	4	1.3s	12.5%	3.5	0.3s
				[0,0.6]	5	1.5s	0.0%	5.0	0.4s
				[0,1.0]	5	1.6s	0.0%	5.0	0.1s
			A	[0,0.4]	5	105.8s	10.0%	4.5	0.4s
				[0,0.6]	4	129.6s	0.0%	4.0	0.5s
				[0,1.0]	5	4.1s	20.0%	4.0	0.4s
			FF	[0,0.4]	5	3.2s	10.0%	4.5	0.5s
				[0,0.6]	5	4.8s	10.0%	4.5	0.5s
				[0,1.0]	5	167.2s	20.0%	4.0	0.4s

Tabela 5.6: Resultados computacionais do modelo Min Max da Engenharia de Tráfego com 10 nodos de serviço.